Guida all'Utilizzo

V1R1M1
Build 126

novembre 2015

WWW.NETACROSS.IT

Creazione di un ambiente per un'applicazione

NetAcross provvede a portare in ambiente browser le applicazioni utente. Per fare questo occorre creare un progetto, cioè predisporre un'area dove verranno memorizzati i dati dei files video; poi occorre creare gli utenti che la utilizzeranno.
Non occorrono altre azioni preventive. NetAcross provvederà alle conversioni dei file video il pagine HTML al primo utilizzo degli stessi. La prima volta si attenderà qualche secondo in più ed una volta effettuata la conversione questa sarà disponibile senza attesa per tutti gli utenti. La stessa cosa avviene se viene successivamente modificato dal programmatore un file video. Al primo utilizzo da parte degli utenti questo viene riconvertito senza che questo appaia evidente all'utilizzatore (eccetto la breve attesa per la conversione).
E' possibile comunque abbreviare il tempo di conversione eseguendo il comando NETDP/CVTDDSLIB fornendogli il nome della libreria su cui operare. Questo comando prepara i files NADDS01F e NADDS02F con i campi e le keyword dei file video in oggetto. Per ogni file video che non è già stato convertito il comando sottomette in batch un lavoro, utilizzando gli attributi della sessione. Questo comando può anche essere schedulato per riallineare i file video compilati nella giornata.
Solo nel caso di utilizzo della modalità batch è necessario prima di iniziare a lavorare compilare i programmi che utilizzano files video con la speciale opzione NetAcross. Si consiglia all'inizio per le prime prove di lavorare nella modalità interattiva per poi passare successivamente alla modalità batch, al fine di semplificare il lavoro iniziale. Il passaggio fra modalità diverse è indolore e tutto il lavoro di conversione effettuato è identico.

I passi necessari ad eseguire un'applicazione nell'ambiente NetAcross sono quindi i seguenti:

- Creazione progetto. Al momento della conversione manuale viene richiesto un file di progetto che indica sulla base della Root in quale applicazione e personalizzazione generare gli archivi. Vedere la sezione Menù Configurazione NetAcross – Funzione SVIL – Gestione Progetti per dettagli circa la creazione.
- Creazione di uno o più utenti NetAcross. Tale utente può corrispondere o ad un utente di sistema (tipicamente uso interno) oppure può essere un utente di Validation List (tipicamente uso Internet). Vedere la sezione Menù Configurazione NetAcross – Funzione CONF – Gestione Iscrizione Utenti per dettagli circa la creazione.

Direttive HTML nelle DDS

L'applicazione prevede di generare delle pagine HTML a partire dai pannelli DSPF. Questa traduzione può essere facilitata inserendo delle direttive aggiuntive nel codice DDS che permettono di dare un significato specifico ai campi AS400 e di utilizzare quindi le funzioni aggiuntive disponibili nelle pagine HTML.
Ad esempio un campo AS400 che prevede la scelta fra una serie di valori (DDS Values) viene trasformata automaticamente in HTML con un menù a discesa e l'elenco dei valori da scegliere. Però ad ogni valore non può corrispondere la descrizione equivalente perché questa non è prevista a livello di DDS. Inserendo una keyword NetAcross nelle DDS è possibile dare le informazioni necessarie alla creazione di un menù a discesa completo di descrizioni.
Le keyword previste possono essere inserite in due modi: nell'area testo del campo interessato (keyword Text) oppure come commento nel sorgente (A* a colonna 6) subito prima del campo interessato.
Se si utilizza la prima modalità si può compilare il file video senza modalità aggiuntive ma si ha l'inconveniente che la keyword text prevede solo 50 caratteri ed occorre provvedere a continuare i dati eventualmente eccedenti su campi hidden come spiegato qui di seguito.
Se si utilizza la seconda modalità si possono scrivere le keyword nel campo commento immediatamente precedente il campo interessato. Occorre indicare A*NA a partire da colonna 6 e scrivere le keyword di seguito fino alla colonna 80. Se occorre andare a capo continuare subito dopo A* a colonna 6. Se occorre forzare il termine dell'analisi del commento scrivere /NA* al termine del commento. Questi commenti sono gestibili anche da SDA.
Esempio:
A*NA*SEL C=Clienti F=Fornitori /NA*
A**MNU v=radioO
Per compilare il file video occorre utilizzare il comando NETDP/CVTDDS. Vedere Opzioni automatiche per PDM nel manuale Installazione e Configurazione se si vuole inserire questa opzione automatica nel PDM. Questo comando prende il commento e provvede a creare le keyword TEXT e i campi Hidden opportuni con le sintassi di continuazione. Poi lancia la compilazione del File Video. Occorre prestare attenzione al programma che gestisce il file video poiché se sono stati modificati con questa operazione dei campi Hidden questi potrebbero generare errori di livello.
Nel caso si utilizzino le keyword inserite nell'area testo (keyword Text) queste devono essere inserite in corrispondenza campo interessato e devono rispettare le seguenti regole di sintassi. Per essere riconosciuta il campo testo deve iniziare con NA nelle prime due posizioni. Il campo testo prevede al massimo 50 caratteri, per cui è possibile che lo spazio sia insufficiente se si prevedono più keyword contemporaneamente oppure nella decodifica della keyword DDS Values. E' previsto un meccanismo di continuazione che prevede la seguente sintassi:
Se l'ultimo carattere di un campo testo contenente keyword NetAcross (NA* nelle prime tre posizioni) è un – (trattino) significa che la riga prevede una continuazione. La continuazione avviene ricercando un campo hidden nello stesso formato record con il nome composto dal nome campo originale + H + <numero>. Tale campo hidden deve avere il campo testo con a posizione uno un trattino. Ad esempio un nome campo CODCLI se ha un trattino a posizione 50 prevede la ricerca di un campo Hidden CODCLIH1 che deve avere un trattino a posizione uno. Se questo a sua volta ha un trattino a posizione 50 viene ricercato CODCLIH2 fino a CODCLIH9. Se la sintassi

non viene rispettata il campo viene considerato come campo testo normale e non keyword NetAcross. Attraverso la keyword *HNA (vedere più avanti) è possibile definire il nome del campo hidden con cui proseguire nella ricerca del testo.
Nel caso di continuazione del campo testo di un tracciato record vengono ricercati campi hidden con la stessa sintassi ma con il nome che spazia da H1 a H9.
Il campo testo risultante può quindi prevedere fino a 432 caratteri. Da tenere conto che gli spazi fra la fine di un campo e l'inizio dell'altro sono significativi.
In un campo testo si possono inserire più keyword. Non occorre ripetere NA ma solo la keyword (ad.es. *SEL). Gli spazi fra keyword non sono significativi.
Per ridurre la complessità del riempimento del campo testo e delle eventuali continuazioni su più campi hidden è prevista la possibilità di inserire le keyword come commento nel sorgente. Occorre inserire NA* a partire da colonna 5 e poi seguire con le keyword. I commenti possono essere inseriti sia con SEU che con SDA. Un apposito comando (NETDP/CVTDDS) di precompilazione provvederà a duplicare il commento nel campo testo e provvederà a creare i campi hidden e concatenarli ove necessario. Poi esegue la compilazione del file video. Vedere Opzioni automatiche per PDM nel manuale Installazione e Configurazione per l'esecuzione facilitata da PDM.

Sono previste le seguenti funzioni:

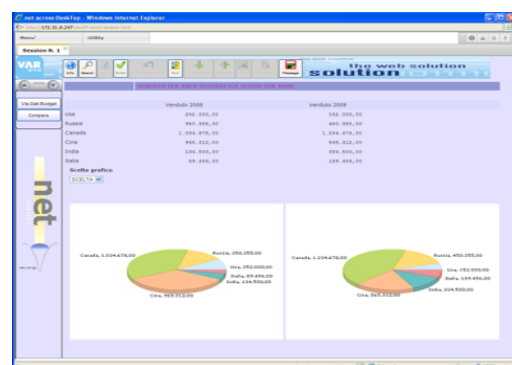
---

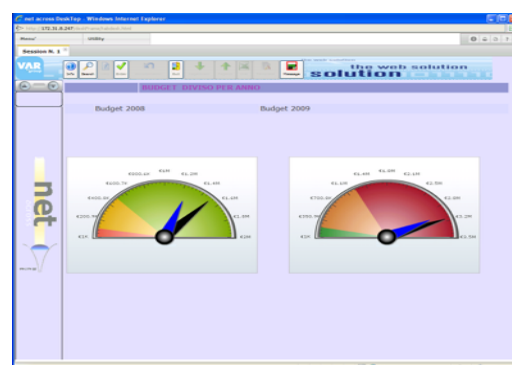
*CHR **(DEPRECATA non sarà disponibile dalle prossime versioni)**

Questa keyword a livello di campo permette di creare dei grafici in formato Flash nella pagina nella posizione del campo su cui è definita. Tale keyword è utile quando si vogliono trasformare o integrare le informazioni numeriche in grafici, istogrammi, cruscotti, ecc.
Come parametri prevede:

t:<tipo grafico>	(testo del pulsante che viene creato)
i:<ID univoco del grafico>	(ID univoco all'interno della pagina per identificare il grafico)
w:<larghezza>	(Larghezza in pixel del grafico)
h:<altezza>	(Altezza in pixel del grafico)
r:<campo di riferimento>	(campo di riferimento)

Il tipo grafico può essere un nome file (fra apici) di tipo .swf presente nella cartella /NetAcross/Scripts/Charts/Charts/*.swf
oppure *NOME: valore speciale tabellizzato nel file NETAP/NACHR00F
oppure [NOMECAMPO] nome campo del video che a runtime conterrà il nome del file o il valore speciale di cui sopra.
I valori dei grafici devono essere contenuti in un file xml nella cartella /NetAcross/Chart dove le sottocartelle sono con la stessa sintassi delle pagine HTML per i formati video.
Es.: /NetAcross/Chart/NetaDemo/Base/Ita/NetaDemo/CHARV.FM03.Pie3D.xml
La sintassi di compilazione del file xml è in:
http://www.netacross.it/aggiornamenti/documentazione/charts/index.html

Esempio:
se si vogliono aggiungere due grafici a torta aggiungere :
NA*CHR T:[CHR001] I:PIE3D W:450 H:350
NA*CHR T:[CHR001] I:PIE3D1 W:450 H:350

Esempio:
se si vogliono aggiungere due grafici a indice aggiungere :
NA*CHR T:"*ANGULARGAU" I:ANGULAR7 W:400 H:300
NA*CHR T:"*ANGULARGAU" I:ANGULAR71 W:400 H:300

---

*CLS

Questa keyword a livello di record o di campo permette di attribuire ad una riga o ad un campo una classe specifica (relativa al file CSS). Tale keyword è utile quando la riga od il campo devono essere evidenziati con colori o attributi particolari, oppure quando la riga od il campo necessitano di un passo non proporzionale (ad esempio quando ho campi incolonnati con significati posizionali). Se la classe indicata è stata definita nel file di riferimento classi CSS, residente nella directory Style, sarà possibile modificare l'aspetto del campo o della riga del video.
A Livello di formato record prevede come parametri :

<classe>	(obbligatorio)
<n.riga>	(uno obbligatorio, altri a seguire facoltativi)

la struttura può essere ripetuta con più classi/righe, separate da almeno uno spazio.

Esempio:
se si vuole attribuire alle righe 5 e 8 una classe che prevede una riga di separazione colorata e alla 10 14 19 20 23 una di un'altro colore (vedere figura sotto) inserire nel campo testo dello stesso:
NA*CLS titolo05 5 8 titolo04 10 14 19 20 23

Nel caso della sintassi per record , il numero di riga è quello assoluto della videata

A livello di campo come parametri prevede:

v:*nome_classe nome_classe nome_classe* … (obbligatorio)

La prima classe indicata è l'unica che sarà sensibile agli attributi del video se presente nella lista delle classi sensibili, impostata sul config.xml. Le altre permettono di gestire particolari attributi grafici come le eventuali funzionalità attivate tramite la keyword *COM

NEW! novembre 2014
E' possibile assegnale la *CLS anche ai campi hidden non di subfile e in questo modo verranno creati dei campi hidden in html con la classe indicata, con proprietà di output dati per essere usati con le funzioni *COM e le funzioni dialog **NetA.confirm** (per default: .CONFIRM) e **NetA.alert** (per default: .ALERT)

Esempio:
se si vuole attribuire ad un campo incolonnato una classe che prevede un passo non proporzionale inserire nel campo testo dello stesso:
NA*CLS v:const

Riferimenti in configurazione per *CLS

CONFIG.XML

```
<ClassNumeric>const</ClassNumeric>: Stabilisce che in fase di conversione
la classe indicata sia utilizzata per i campi numerici, si usa una classe
con font a passo constante per mantenere l'allineamento dei caratteri.

<ClassesWithAttribute>
 <Class>const</Class>
 <Class>normal</Class>
 <Class>nasfc_class</Class>
</ClassesWithAttribute>
L'area indica una lista di nomi classi che verranno "declinate" secondo la
regola della tabella degli attributi/colori. Si potrà quindi stabilire una
caratteristica grafica in base agli attributi del video presenti sul campo
se la classe impostata sarà una di quelle indicate nella lista

<ClassProtect>"color: #868686;border: 2px hidden;background-color:
#DFDFDF;font-weight: bolder;"</ClassProtect>
Questa definizione non è una classe ma un insieme di attributi di stile
che saranno utilizzati in sovrapposizione alle regole di classe del campo
quando il campo è protetto sul display file.

<ClassError>error01</ClassError>
Questa tag indica la classe che sarà utilizzata quando per il campo (input
solitamente) sarà acceso un indicatore di errore

<WidthMultInfo>
 <WidthMultCFac>normal=8,0</WidthMultCFac>
 <WidthMultCFac>input01=8,0</WidthMultCFac>
</WidthMultInfo>
Questa tag permette di indicare un fattore di proporzione per gli spazi
bianchi sostitutivi dei caratteri in base alle classi impostate,viene
utilizzato, per esempio,per mostrare un numero sufficiente di caratteri a
passo variabile in un campo di input.
```

*CMD

NetAcross crea automaticamente una pulsantiera per la gestione dei tasti di comando legati alle pagine video. Quali tasti comando sono abilitati viene rilevato automaticamente. La descrizione da associare ad ogni tasto non è invece un dato rilevabile dal file video. Viene quindi scansionato in sede di generazione il file video alla ricerca delle descrizioni. Questo viene effettuato facilmente se i tasti di comando sono rappresentati come campi costanti. Se invece vengono utilizzati dei campi di output riempiti a runtime (tipicamente nelle applicazioni multilingue) non si è in grado di

riempire correttamente la pulsantiera e viene inserito una descrizione generica NO TEXT.
Questa keyword a livello di campo permette di indicare al generatore di Html che la costante o il campo di output rappresentano una descrizione di tasto di comando e che il contenuto può essere utilizzato per riempire le descrizioni della pulsantiera. Il campo una volta portato sulla pulsantiera viene eliminato dall'Html.
E' utilizzabile anche sui campi nascosti (solo modalità batch).
Questa keyword si differenzia dalla keyword *CMR perché prevede la ricerca di un unico tasto di comando dentro il campo. In alcune applicazioni vengono utilizzati una serie di campi di output lungo tutta la riga e questi vengono riempiti a tempo di esecuzione con i tasti di comando attivi in funzione delle circostanze specifiche. Se usata la keyword *CMR inibisce la keyword *CMD.
Come parametri prevede:

v:<tasto comando>="<stringa>"	(obbligatorio. Possono essere inseriti n valori separati da spazio. Definisce l'associazione a tasto comando. La stringa (opzionale) viene ricercata a runtime nel campo per vedere se è valida la condizione.)

Esempio:
se un campo di output conterrà a runtime la stringa F4=Annulla espressa nelle varie lingue indicare nel campo testo dello stesso:
NA*CMD v:F4="F4="
Esempio:
se un campo di output conterrà a runtime la stringa F4=Annulla oppure F16=Ripristina indicare nel campo testo dello stesso:
NA*CMD v:F4="F4=" F16="F16="
Esempio:
se più campi costante sovrapposti contengono la stringa F4=Annulla espressa nelle varie lingue indicare nel campo testo degli stessi:
NA*CMD v:F4

*CMR

NetAcross crea automaticamente una pulsantiera per la gestione dei tasti di comando legati alle pagine video. Quali tasti comando sono abilitati viene rilevato automaticamente. La descrizione da associare ad ogni tasto non è invece un dato rilevabile dal file video. Viene quindi scansionato in sede di generazione il file video alla ricerca delle descrizioni. Questo viene effettuato facilmente se i tasti di comando sono rappresentati come campi costanti. Se invece vengono utilizzati dei campi di output riempiti a runtime (tipicamente nelle applicazioni multilingue) non si è in grado di riempire correttamente la pulsantiera e viene inserito una descrizione generica NO TEXT.
Questa keyword a livello di campo permette di indicare al generatore di Html che la il campo di output rappresenta più descrizioni di tasto di comando e che il contenuto può essere utilizzato

Riferimenti in configurazione per le *CMD e *CMR

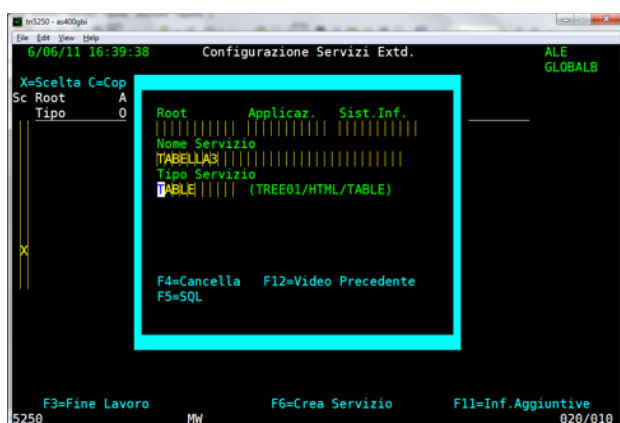
COMMANDS_DEFAULTS.JS

```
var KEYFVIEW = [Y/N]
Rende visibile la "Fxx" nel testo descrittivo del tasto funzionale.

var DELETE_BLANK_KEYF = [Y/N]
Elimina dal pannello dei comandi i tasti funzionali attivi ma senza una
descrizione esplicitata a video (tasti attivi ma nascosti)

var CMD_*
Tutte le variabili CMD_ di definizione grafica del pannello dei comandi
```

per riempire le descrizioni della pulsantiera. Il campo una volta portato sulla pulsantiera viene eliminato dall'Html.
Questa keyword si differenzia dalla keyword *CMD perché prevede la ricerca di più tasti di

Riferimenti in configurazione per le *CMD e *CMR

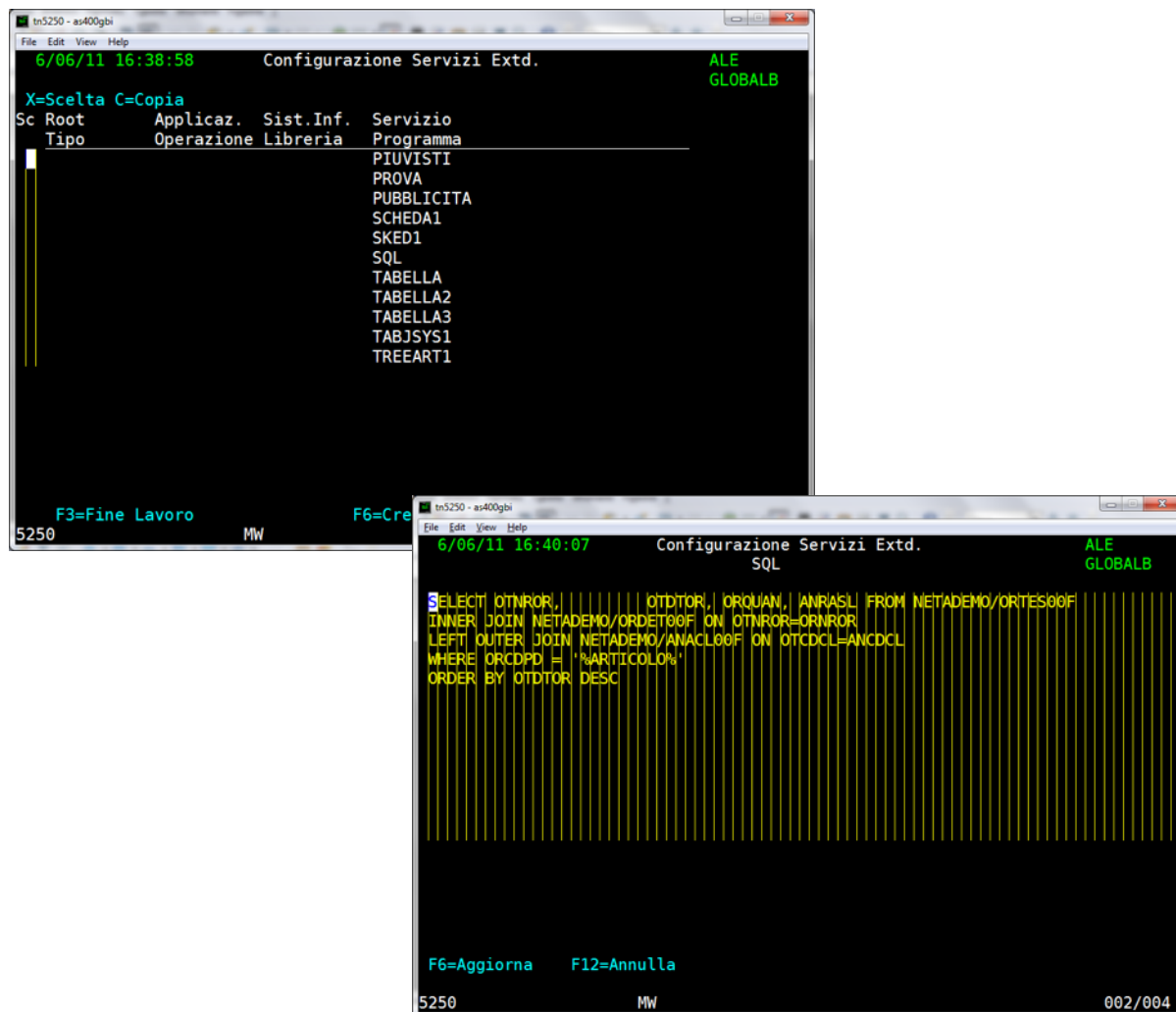
CONFIG.XML

```
<KeyFSplit>[Y/N]</KeyFSplit>: Attiva la ricerca di più tasti funzionali
nelle costanti del video, se "N" si ferma alla prima. Se si usa *CMD o
*CMR il valore è da tenere a "N"

<KeyFunctions>
 <Pattern>F[0-9]+[0-9]*=.+ </Pattern>
 <Pattern>F[0-9]+[0-9]* .+ </Pattern>
 <Pattern>CMD[0-9]+[0-9]*=.+</Pattern>
 <Pattern>CMD[0-9]+[0-9]*-.+</Pattern>
 <Pattern>F[0-9]+[0-9]*/.+</Pattern>
</KeyFunctions>
L'Area KeyFunctions stabilisce le regole di riconoscimento del testo come
definizione di tasto funzionale (massimo 9 pattern) tramite espressioni
regolari.
```

comando dentro un unico campo. In alcune applicazioni viene utilizzato un unico campo di output lungo tutta la riga e questo viene riempito a tempo di esecuzione con i tasti di comando attivi in funzione delle circostanze specifiche.
E' utilizzabile anche sui campi nascosti (solo modalità batch).
Se usata la keyword *CMR inibisce la keyword *CMD.
Dal punto di vista delle prestazioni tenere conto

che la ricerca dei tasti di comando viene effettuata non in fase di conversione Html ma in fase di esecuzione utilizzando le espressioni regolari indicate della keyword <Pattern> del file Config.xml. La keyword non prevede parametri.

Esempio:

se un campo di output conterrà a runtime la stringa F4=Annulla e F16=Ripristina indicare nel campo testo dello stesso:

NA*CMR

*COM

Descrizione

La keyword permette di aggiungere alle pagine html generate dai display files alcuni ***componenti*** (da qui il nome) grafici e funzionali. Al momento sono previste 4 tipologie di oggetti con diversi parametri e possibilità di estenderne le funzioni: tabelle, liste ad albero, html puro e ricerche contestuali.
Questi oggetti portano con se delle funzionalità “esterne” alla pagina, collegate in modo logico agli elementi del display file. Si compongono di una parte grafica e di una parte applicativa che è asincrona rispetto al flusso del programma, di fatto il programma non “vede” i *componenti* ma questi possono interagire con la pagina.
La comunicazione asincrona permette di svincolarsi dal flusso pagina per pagina che i dati devono seguire nell’interfaccia la programma RPG, i componenti si alimentano e generano eventi in maniera *non bloccante* per il flusso del programma.

E’ possibile utilizzare alcuni campi della videata come parametri per i *componenti* ed altri come ricevitori delle azioni dell’oggetto.
Un componente può essere posizionato su un campo della videata, come di solito si fa con le altre keyword, **oppure esternamente alla FORM** che contiene i dati attivi che il programma riceve dal browser, in aree predisposte per ricevere tali oggetti.

Le aree esterne sono ottenute utilizzando un templare dello scheletro html specifico, un template di default è disponibile come template.COM.html nella cartella /CONFIG, e nel caso ci sia definita almeno una keyword *COM in uno dei formati compresi in un html , il convertitore utilizzerà questo template al posto dello standard.

Il **template.COM.html** contiene sia html standard che dei marcatori necessari in fase di conversione per identificare le varie aree del documento, questo è un esempio:

```
<body bgproperties="fixed" bgcolor="#DEDEE6" leftmargin="0" topmargin="0" marginwidth="0"
marginheight="0">
<!--NA_MAIN_WIDTH:700-->
<!--NA_IGNORE_IN-->
<div id="main_all" align="left">
<div id="main_top" style="position: relative; width: 960px; top: 0px; display: none;"></
div>
<div class="bodycontent" style="width: 960px;">
<div id="main_left" align="left" style="float: left;  height: 100%; width: 180px;
display: none;">
</div>
<div id="main_center" align="left" style="overflow: hidden;float: left; width: 700px;">
<!--NA_IGNORE_OUT-->
<form name="modulo" action="/cgi-bin/NET_ACROSS.PGM/<PERSIST>" method="POST"
nawidth="700">
<table width="100%" border="0" cellspacing="1" cellpadding="0"  height="400"
class="sfondo01">
        <tr>
          <td height="77" class="TDmenu" valign="top">
<na_screen />
          </td>
        </tr>
      </table>
```

```
</form>
<!--NA_IGNORE_IN-->
</div>
<div align="left" id="main_right" style="float: left;  height: 100%; width: 80px;
display: none">
</div>
</div>
</div>
<!--NA_IGNORE_OUT-->
</body>
```

Le nuove aree sono identificate da id="main_*" , è da notare che il blocco identificato da id="main_center" è quello che contiene la struttura minimale del template originale.
Come si vede ci sono dei marcatori (commenti con NA_IGNORE) che stabiliscono le parti personalizzabili, il convertitore le ignorerà, potranno quindi essere aggiunte o tolte altre aree, impostate le loro dimensioni e posizione.

Configurazione Servizi

Ad ogni **componente** è associato un **servizio** che ne identifica la tipologia e stabilisce le regole.

Per configurare un servizio:
CALL NETAP/NETAMENU CONF + Opz.27

L'opzione permette di gestire il file di configurazione dei servizi, la chiave del file è simile a quello degli alias con una chiave alfabetica di 32 caratteri che identificano univocamente il servizio.
L'identificazione del servizio avviene in base ai dati di chiave, se i valori sono lasciati in bianco vale per tutte le istanze/applicazioni/sistemi informativi

Root	è l'istanza di riferimento
Applicazione	il nome applicazione come configurato sul profilo utente
Sist.Inf	nel caso si usi un'architettura è possibile identificare un servizio con lo stesso nome ma costruito in maniera diversa per sistema informativo diverso.
Nome Servizio	Il nome servizio che verrà inserito nei display files
Tipo Servizio	Sono abilitati 4 valori per il servizio TREE01 / SQL / HTML / SEARCH
Programma Exit	Nel caso di TREE01 e HTML è necessario inserire il nome del programma che fisicamente si occupa di gestire la logica di riempimento dell'oggetto.

I servizi si distinguono per due principali categorie, quelli di tipo **CALL** che sono attualmente il **TREE01** e l'**HTML** e quelli di tipo **SQL** che sono la **TABLE** e la **SEARCH**.

Nel caso di TABLE e SEARCH è disponibile un tasto funzionale **F5** per gestire un SQL parametrico che ritorna i dati di riempimento dei rispettivi oggetti.

Il servizio net-across si occupa di eseguire il programma e l'sql in un job con la **lista librerie** del job applicativo che l'ha richiesto, e con il **profilo utente attivo** per quella sessione, se il job principale ha delle **override attive** su file di database utilizzati, **queste non saranno disponibili**.

Keyword e parametri DDS

La *COM si inserisce sempre a livello di formato record, anche nel caso sia agganciata ad un campo della videata, quindi potranno essere inserite 3 o 4 keyword per formato record in base al numero di caratteri utilizzati (max 400 caratteri)

Come parametri si prevedono:

s:<nome_servizio>	E' il nome del servizio come presente nell'opzione 27 del menù CONF (Obbligatorio)
f:<nome_campo>	Indica il campo valorizzato dal componente o valore di ritorno, nel caso di ambiente batch possono essere usati anche campi hidden (Facoltativo)
t:<titolo>	se il componente lo prevede sarò utilizzato

	nell'intestazione dello stesso , deve essere espresso fra apici "TITOLO" (Facoltativo)
a:<attach>	è il punto di aggancio del componente, segue la sintassi css con **#id_elemento** o **.classe_elemento**, se non indicato equivale a **#main_left** (Facoltativo)
p:<params>	elenco di coppie nome/valore passati come parametri al servizio, separati da virgola composti come { NOME : '%CAMPO%' } (Facoltativo)
d:<data>	parametri di configurazione specifici per componente (attualmente disponibili solo per l'oggetto TABLE). (Facoltativo)

Parametro a:<attach>
Il punto di aggancio fisico del componente, determina quindi al posizione dell'oggetto grafico nella pagina html, la sintassi prevede di identificare l'elemento tramite la sintassi CSS quindi è possibile indicare una DIV presente nel *templare.COM.html* utilizzato indicando "**#**" seguito dal parametro **"id"** dell'html. Il default posiziona l'elemento nell'area identificata da "**#main_left**" che, come indca il nome, nel templare base è posizionata a sinistra del formato video.
La sintassi CSS permette anche di identificare una classe con la sintassi ***.nome_classe*** in modo da poter agganciare quei campi che sono identificati con la keyword *CLS, il valore del parametro è sensibile alle maiuscole e minuscole per l'identificazione della classe.

Parametro p:<params>
I parametri passati al servizio sono impostati come lista di valori con sintassi:
aperta graffa "{" + nome del parametro come sarà visto nel servizio + due punti ":" + valore del parametro come valore alfabetico o come riferimento a un campo della videata indicato con percento+nome_campo+percento.
Esempio:
```
p:{ARTICOLO:'%WWSKE%',FASE:'1'}
```

il servizio riceverà il parametro **ARTICOLO** con valore preso dalla videata nel campo **WWSKE** e il parametro **FASE** con valore fisso **1**.

Parametro d:<data>
Al momento solo la TABLE utilizza questi dati, la sintassi degli stessi dipende di fatto dal componente e la trattazione della sintassi non è disponibile in questo manuale.

Esempio:
Questo è un esempio di DDS sorgente in si trovano vari esempio di utilizzo dei parametri

```
A*NA*JSC /SCRIPTS/PROVA.JS
A* *COM S:TREEART1 F:WWRET T:"Scelta Articoli" A:#main_left
```

```
A* *COM S:SKED1 F:WWRET T:"Scheda" A:.S1
A* *COM S:TABELLA3 F:WWORD T:"Ordini Relativi" A:.T1 P:{ARTICOLO:'%WWSKE%'}
A*    D:{onCellSelect: provaDemo,height:120,width:458}
A          R FORM1
A*%%TS  SD  20110525  163008  ALE         REL-V5R2M0  5722-WDS
A                                      CA03
A                                      SETOF(90)
A                                  1  2'Visualizzazione Articoli'
A                                      DSPATR(HI)
A            WWSKE2        11A  H
A            WWORD         30A  H
A            WWRET         11A  H
A*NA*CLS V:T1
A                                  4  2'T1'
A                                 22 37'F3=Fine'
A                                      COLOR(BLU)
A                                 22 37'F3=Fine'
A                                      COLOR(BLU)
A*NA*IMG p:Y V:"/imgdemo/"[WWRET](11)".jpg" H:220  *CLS V:imgsk
A                                  3  2'I1'
A*NA*CLS V:S1
A            WWSKE         11A  O  3 10
```

Parametri di Input e Output

I servizi come detto si dividono per tipologia in CALL e SQL, nel caso di CALL l'interfaccia del programma di exit è un campo di output di 64000 caratteri che deve essere appunto riempito dal programma di exit.
Nel caso di HTML il riempimento è libero, si deve cioè produrre un html formalmente corretto, mentre nel caso TREE01 c'è una DS la TREE01DS che deve essere usata come "maschera" di riempimento dell'area (vedi nello specifico)

Per ricevere i parametri di input è necessario usare un programma di servizio la **NASRVRD** che ritorna il valore contenuto nel parametro passato.
Il parametro **ReqVar** è il nome del parametro come predisposto sulle DDS , (256 A)
Il parametro **ReqLgh** è la lunghezza massima della variabile predisposta a leggerne il contenuto (5S 0)
Il parametro **ReqVal** è l'area che verrà valorizzata con il valore del parametro per gli n caratteri impostati. (32760 A)

Esempio:
Recupero il valore del parametro "value" prendendo al massimo 50 caratteri

```
Call     'NASRVRD'
Parm     'value'      ReqVar        256
Parm     50           ReqLgh          5 0
Parm     *blanks      ReqVal      32760
```

La chiamata al NASRVRD può essere effettuata all'inizio della exit per ogni variabile (parametro) definita nel parametro P: della keyword *COM delle DDS. In pratica per ogni servizio CALL è il programma associato che recupera i dati passati nella definizione della keyword.

Nel caso si usi il nome di parametro **"value"** (in minuscolo) si potrà recuperare un valore di

riferimento specifico per l'oggetto, quindi dalla pagina HTML origine.

Nel caso del componente, per **TREE01** è il valore di ramo selezionato (vedere più avanti).
Nel caso di **HTML** (se disponibile) è il valore "value" del campo su cui è posizionato l'oggetto.
Nel caso **SQL** il campo non è disponibile.
Nel caso **SEARCH** il campo "VALUE" (maiuscolo) è utilizzabile nell'SQL di ricerca.

Nel caso SQL è possibile utilizzare i parametri nella stringa SQL se espressi con ***%nome_parametro%*** nel caso la stringa SQL risultante fosse errata verrà segnalato un errore non bloccante.

I Tipi

TREE01

Il componente disegna un menù ad albero dove ogni foglia può essere una azione o un menù o volendo un file e una cartella.

L'albero si comporta in questo modo:quando l'utente seleziona un ramo e questo non è visibile (cioè è presente il simbolo **"+"**), viene richiamato il server passando nel parametro **"value"** il valore indicato nell'elemento scelto e il ramo scaricato non verrà più aggiornato fino alla prossima ricarica completa della pagina.

Nel caso l'utente preme una foglia "azione" allora il valore delle foglia viene scritto in un campo della videata (configurato tramite la keyword sulle DDS) e viene premuto **"Invio"**, quindi il controllo torna al programma RPG principale che può analizzare la scelta dell'utente.

Il programma exit ha un unico parametro di output di 64000 (tipo CALL) che deve essere riempito con n ripetizioni della DS TREE01DS , in cui ogni elemento è una foglia del ramo da rappresentare, in pratica si riempie un livello per volta per chiamata.

La **TREE01DS** è così composta:

T01TYPE = 'A' o 'M' per indicare se la foglia è Azione o Menù (sinonimo di File o Cartella)

T01VALU = 'valore' un codice alfabetico di 32A che sarà usato come valore di ritorno per il programma se il ramo è 'A' , oppure se 'M' sarà il parametro **value** che sarà restituito alla exit per caricare un nuovo ramo.

T01DESC = 'Testo di descrizione' un testo di 32A come descrizione della foglia

T01CLAS = '*class-name*', un nome di una classe da usare per rappresentare (icona/colore) la foglia, secondo le regole disponibili nel file del foglio di stile di default per l'albero che si trova in ***/netacross/deskFrame/jquery/plugins/jquery.treeview.mod.css***

E' possibile aggiungere nuovi stili e immagini per creare rami personalizzati utili a segnalare diverse tipologie di dati gerarchizzati.
Questo tipo di componente è predisposto per essere posizionato fuori dall'area del DSPF in quanto la sua dimensione è variabile e non facilmente calcolabile a priori per cui è necessario che abbia una sua area di lavoro indipendente dal video.

Esempio:
sarà creato un albero del servizio TREEART1 con valore di ritorno a programma nel campo WWRET,definito hidden, un titolo descritto e posizionato come di default a sinistra dell'area principale

```
A* *COM S:TREEART1 F:WWRET T:"Scelta Articoli" A:#main_left
..
A            WWRET         11A  H
```

Il programma associato al servizio è in **NETAP/QRPGLESRC** con nome **PROVASV1**

HTML

Il parametro di output è di 64000 caratteri e deve essere riempito con HTML ben formattato (non viene effettuato nessun controllo formale), il codice HTML composto andrà a sostituire il campo della videata a cui è agganciata la funzionalità, il parametro “**value**” in questo caso è il valore del campo stesso, se l'oggetto HTML non è posizionato su un campo questo valore non sarà disponibile.

Questo componente può essere usato per inserire in una parte del video, come i campi di un subfile, un codice html graficamente più piacevole, che può contenere anche immagini e link esterni, con una densità di informazioni maggiore di quella disponibile sul display file.

Esiste anche la possibilità di **rendere attivo** , cioè sensibile al click del mouse, il componente:

Se nel html prodotto è presente una **TAG** di tipo con un parametro con nome **value** (es.: ...codice...) automaticamente l'area contenuta diventa sensibile al click e può valorizzare il campo di ritorno della keyword DDS con il valore indicato, se presente un attributo **fkey** è possibile simulare anche un tasto (verrà richiamata la della NetA.F(”xx”))
Sarà possibile anche gestire eventi diversi con l'attributo **action** valorizzato, per default l'evento sarà **onclick**.

Esempio:
Sarà aggiunto un componente del servizio SKED1 con valore di ritorno WWRET e posizionato al posto del campo che ha come classe il valore **“S1”** , più sotto si vede la definizione della classe per il campo di riferimento, essendo questo un servizio HTML, sarà disponibile il parametro **“value”** valorizzato con il valore del campo WWSKE al momento dell'emissione del video.

```
A* *COM S:SKED1 F:WWRET T:"Scheda" A:.S1
…
A*NA*CLS V:S1
A            WWSKE        11A  O   3  10
```

Il programma associato al servizio è in **NETAP/QRPGLESRC** con nome **PROVAHTM1**

TABLE

Verrà generata una tabella con i risultati dell'SQL impostato. Se i campi dei file selezionati hanno la COLHDG definita verrà usata questa come descrizione della colonna.
Nel caso in cui venga usata la sintassi **SELECT NOMECAMPO AS MYALIAS**, il valore **MYALIAS** verrà usato questa come intestazione della colonna e come nome del campo della tabella locale.
E' possibile rendere variabile l'SQL con la sostituzione di testo identificato da %NOMECAMPO% con il valore nel parametro ricevuto con lo stesso nome, i nomi campo disponibili saranno quelli

definiti nel parametro **P:** della keyword *COM
La tabella può leggere campi dal video (al momento non dal subfile) e usarli come variabili dell'SQL inserito in configurazione, ed è possibile rendere attivo ogni valore nelle celle della tabella che se clikkato può valorizzare il campo di ritorno indicato nella keyword, sono disponibili delle funzioni javascript per recuperare i dati dalla riga.

Esempio

Sarà disegnato un oggetto del servizio TABELLA3 che è una tabella con il campo WWORD come ricevitore del valore di ritorno,definito più sotto come campo hidden, agganciato al campo con la classe "T1" che in questo caso è un marcatore di posizione, un campo di output che non verrà visualizzato. La funzione riceve un parametro di nome **ARTICOLO** che conterrà il valore del campo WWSKE della videata, e questo sarà utilizzabile dall'SQL.
In più sono definiti gli attributi della tabella relativi alle dimensioni e alla gestione dell'evento di selezione della cella.

```
A* *COM S:TABELLA3 F:WWORD T:"Ordini Relativi" A:.T1 P:{ARTICOLO:'%WWSKE%'}
A*   D:{onCellSelect: provaDemo,height:120,width:458}
…
…
A            WWORD          30A  H
A*NA*CLS V:T1
A                                4  2'T1'
```

Il servizio (MENU CONF Opz.27) ha un SQL definito in questo modo:

```
SELECT OTNROR,  OTDTOR, ORQUAN, ANRASL FROM NETADEMO/ORTES00F
INNER JOIN NETADEMO/ORDET00F ON OTNROR=ORNROR
LEFT OUTER JOIN NETADEMO/ANACL00F ON OTCDCL=ANCDCL
WHERE ORCDPD = '%ARTICOLO%'
ORDER BY OTDTOR DESC
```

Quindi la stringa %ARTICOLO% verrà sostituita con il valore ricevuto che è appunto il valore del campo del video.

Il parametro **D:** permette di applicare una serie di parametri alla tabella, al momento sono supportati: **width**, **height**, **onCellSelect**, **filter**.

I parametri **width** e **height** impostano le dimensioni della tabella.

Il parametro **filter** abilita una riga in testata dove è possibile eseguire un filtro dei dati in colonna.

L'evento **onCellSelect** è disponibile per identificare la selezione di una cella, per default l'evento valorizza il campo di ritorno con il valore della cella. Nel caso la funzione **provaDemo** è disponibile nella stessa pagina tramite una keyword *JSC.
Questa è la definizione della funzione.

```
function provaDemo(rowid,
          iCol,
          cellcontent,
          e){
          var grid = getJQGrid(e);
          var row = $(grid).jqGrid("getRowData",rowid);
          var codice = row.OTNROR;
          var returnField = getJQreturnField(e);
          $("input[name=" + returnField + "]").attr("value",codice);
          NetA.F('Invio');
     }
```

La funzione riceve 4 parametri: numero riga, numero colonna, il contenuto della cella e l'evento. Utilizza l'evento per recuperare in modo sicuro i dati, prima recupera l'oggetto griglia:
var grid = getJQGrid(e);
poi recupera l'oggetto riga dal numero riga
var row = $(grid).jqGrid("getRowData",rowid);
dalla riga si può recuperare i campi con il nome del campo sulla tabella
var codice = row.OTNROR;
tramite una funzione interna (dell'oggetto TABLE di netacross) recupera il campo della videata impostato come campo di ritorno e tramite jquery imposta il valore
$("input[name=" + returnField + "]").attr("value",codice);
poi preme invio
NetA.F('Invio');

Al momento altri parametri disponibili sono le dimensioni della tabella che per default sono {width: 100% , height: 200px} , ci sono molti altri parametri e eventi gestibili dall'oggetto griglia e verranno gestiti e implementati progressivamente.

E' possibile inserire una immagine in una cella utilizzando una sintassi SQL.
E' necessario definire un campo tramite l'istruzione SQL AS con nome **IMG**+ **3 cifre** [indicanti le dimensioni della colonna tali da poter contenere l'immagine]+"_" [carattere underscore] + "**TITOLO DELLA COLONNA**".

Esempio:

In questo esempio si vede il primo campo come viene definito per generare un immagine (occhietto) nella tabella "Ordini Relativi", sopra definita.

```
SELECT '/IMAGES/EYE.GIF' AS IMG024_CAR,OTNROR, OTDTOR, ORQUAN, ANRASL
FROM NETADEMO/ORTES00F
INNER JOIN NETADEMO/ORDET00F ON OTNROR=ORNROR
LEFT OUTER JOIN NETADEMO/A
NACL00F ON OTCDCL=ANCDCL  WHERE ORCDPD= '%ARTICOLO%'
ORDER BY OTDTOR DESC
```

IMG024_CAR: genera un'immagine larga 24 pixel con titolo CAR (il titolo sarà espresso sempre in caratteri maiuscoli)
Nel valore del campo è passato il path relativo (senza /netacross) dell'immagine, quindi in questo caso un valore fisso.

SEARCH

La SEARCH genera un widget "autocomplete" in cui il testo scritto viene inviato al servizio che riproduce in formato lista la risposta. In pratica il testo inserito da un utente in un campo di input è utilizzabile come valore parametro in un SQL con le stesse modalità della TABLE ma i cui valori di ritorno hanno un significato ordinale ben preciso:

Il primo campo ritornato dal SQL è il valore di confronto, il secondo se disponibile è il valore di ritorno, altrimenti verrà usato il primo campo anche come valore di ritorno.
Il campo **%VALUE%** inserito nell'SQL contiene il testo inserito dall'utente nel campo indicato dal parametro **F:<nome_campo>**
Il componente scatena il suo riempimento ogni volta che il contenuto del campo di input è modificato dall'utente, se necessario filtrare i dati per un numero maggiore di caratteri è necessario usare per esempio **D:{minChars: 3}**

Esempio:
Ci sono 2 campi di ricerca agganciati a 2 servizi diversi, il servizio **CERCA** ha questo SQL, in cui il valore viene usato come stringa di confronto interna alla stringa.

```
SELECT DRALIA,DRPATH FROM NETAF/NADIR00F WHERE DRALIA LIKE '%%VALUE%%'
ORDER BY DRALIA
```

Il **CERCA2** ha questo SQL:

```
SELECT DISTINCT D1NFIL, TRIM(D1NLIB) CONCAT '/' CONCAT D1NFIL AS VALUE
FROM NETAF/NADDS01F WHERE D1NFIL LIKE '%%VALUE%%'  ORDER BY D1NFIL
FETCH FIRST 10 ROWS ONLY
```

Come si vede limitare il numero di record in uscita è buona norma su file molto grandi in quanto ad ogni richiesta viene eseguito nuovamente l'SQL.

Sul display file questa è la definizione dei campi:

```
A*NA*COM S:CERCA T:"Cerca Alias:" A:.C1 F:V1_OPZ01
A*   *COM S:CERCA2 T:"Cerca File:" A:.C2 F:V1_OPZ02
A           R INIZIO
A*%%TS  SD  20110629  110827  ALE         REL-V5R2M
A                                  7  9'FORMATO INI
A                                      RD'
A*NA*CLS V:C1
A                                 11  9'I'
A             V1_OPZ01      30A  B 16 10
A*NA*CLS V:C2
A                                 11 39'I'
A             V1_OPZ02      30A  B 16 40
```

HTABLE [NEW! ottobre 2014]

NEW! ottobre 2014
La HTABLE permette di creare un set di tabelle in formato html da inserire nella pagina applicativa. E' necessario implementare una CALL ad un programma con parametro buffer, ma supera il limite dei 64k infatti il parametro è al momento ignorato e lasciato per sviluppi futuri.
Il limite è superato con una struttura dinamica che deve essere generata dal programma stesso tramite alcune API di servizio.
La keyword può generare un set di tabelle a numero di colonne variabili con la possibilità di gestire la formattazione cella per cella.
Tecnicamente il programma guida del servizio deve eseguire le solite operazione per recuperare i

parametri, può quindi costruire le tabelle con questa sequenza:

Inizializzazione dell'area di HTABLE

```
CALL      'NASRVHTA3'
```

Deve essere chiamata una sola volta prima della costruzione delle tabelle, senza parametri.

Definizione della tabella

```
CALL      'NASRVHTA5'
Parm                    TableTitle
Parm                    TableCount
Parm                    NumColumns
Parm                    Columns
```

Deve essere richiamata per ogni nuova tabella che costituisce il set della HTABLE, se c'è già una tabella in costruzione chiude la tabella attiva e apre una nuova tabella.

I parametri sono così definiti:

```
D  TableTitle     S            100A
D  TableCount     S              3S 0
D  NumColumns     S              3S 0
D  Columns        DS
D   Col_Label                  100A   Dim(200)
D   Col_LabCls                  30A   Dim(200)
D   Col_DftCls                  30A   Dim(200)
```

TableTitle: definisce la descrizione della tabella
TableCount: deve contenere il progressivo di conteggio delle tabelle da 1 a 999
NumColumns: contiene il numero di colonne di cui è costituita la tabella
La **DS Columns** contiene il default per le singole colonne
Col_Label: è la descrizione della colonna
Col_LabCls: è la classe di default per la descrizione della colonna
Col_DftCls: è la class di default per i valori del campo contenuto nella colonna

Definizione delle righe

```
CALL      'NASRVHTA7'
Parm                    RigaN
Parm                    NumColumns
Parm                    Flds
```

Può essere richiamata dopo la definizione di una tabella per ogni riga che costituisce la tabella
La definizione dei parametri prevede:

```
D  NumColumns     S              3S 0
D  Flds           DS
D   Fld_Value                  256A   Dim(200)
```

```
D   Fld_Class                     30A   Dim(200)
D RigaN           S               10S 0
```

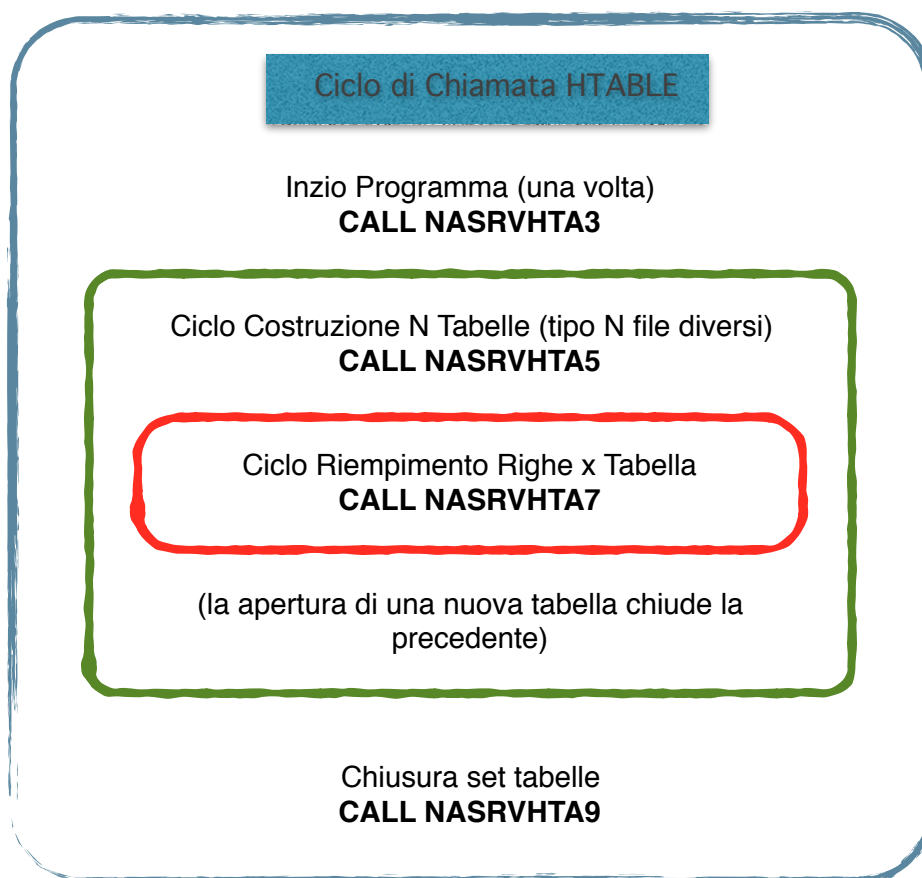
RigaN: deve contenere il numero riga attualmente inviata da 1 a 999999999 (teorico)
NumColumns: deve contenere il numero di colonne passate, equivale al numero indicato nella definizione della tabella
La **DS Flds** contiene i valori delle celle nella riga dove:
Fld_Value: è il valore del campo alfabetico, se rappresentato un numerico la formattazione deve essere fornita o a programma o tramite css
Fld_Class: è il valore di classe per la singola cella, se non fornito verrà utilizzato il valore di default previsto nella definizione della tabella.

Chiusura del set di tabelle

```
CALL      'NASRVHTA9'
```

Deve essere chiamata come ultima operazione del programma di servizio, chiude il set e rilascia il flusso in emissione

Il **numero massimo di tabella** per HTABLE è **999 (solo teorico)**
Il **numero massimo di colonne** per tabella è **200 (con css default)**
Il **numero massimo di righe** per tabella è **999999999 (solo teorico)**
Il **numero massimo di caratteri** per valore nella cella è **256 char**
Il **numero massimo di caratteri** per identificare la classe nella cella è **30 char**

Il numero massimo di tabelle e righe dipendono dalla capacità dell'HTTP Server e dalla gestione della memoria dello stesso, non è possibile stabilire dati certi sull'effettiva dimensione.

SELECT [NEW! giugno 2015]

NEW! giugno 2015

La SELECT permette di creare una lista di opzioni principalmente con scelta multipla (ma anche a scelta singola) in cui il numero di opzioni possibili è determinato a runtime.
Dall'exit è possibile recuperare il valore del campo su cui è posizionato l'oggetto con il campo **value** come già visto per le altre tipologie di *COM.
E' necessario recuperare anche il nome del campo di ritorno, informazioni che sul video è presente e deve essere passata all'oggetto con il campo speciale **returnField.**
Il campo di ritorno può essere anche un campo HIDDEN della videata e nel caso di multi-select conterrà i valori delle opzioni selezionate dall'utente, allineati sinistra. Il campo di ritorno dovrà avere una dimensione sufficiente a contenere i valori previsti selezionabili, altrimenti il valore verrà troncato allo spazio disponibile.

Il programma di exit dovrà richiamare una sola volta il programma **NASRVSEL1**:

```
  *
C                     Call       'NASRVSEL1'
C                     Parm                      returnField       20A
C                     Parm       'S'            multiple           1A
C                     Parm       10             vsize              5S 0
C                     Parm       3              pad                3S 0
  *
```

Questa API riceve il nome del campo di ritorno come recuperato in precedenza dalla videata

“**returnField**”, una flag per indicare se la SELECT sarà a scelta multipla “**mutiple**” e il numero di opzioni visibili “**vsize**” (significativo solo se multiple = ‘S’)
Il parametro “**pad**” indica il numero di caratteri massimo nei valori delle opzioni, il campo di ritorno conterrà questi valori scelti, allineati con questa distanza uno dall’altro.

E successivamente dovrà richiamare il programma **NASRVSEL2** per ogni opzione da visualizzare:

```
 *
C                    Call      'NASRVSEL2'
C                    Parm                   value          64
C                    Parm                   descr         256
C                    Parm                   selected        1
C                    Parm                   disabled        1
C                    Parm                   group         256
C                    Parm                   last            1
 *
```

Dove **value** è il valore dell’opzione e **descr** la descrizione come sarà visibile dall’utente.
La flag **selected** evidenzierà i valori già selezionati se il valore passato è diverso da *blank.
La flag **disabled** renderà l’opzione visibile ma non selezionabile se passato diverso da *blank.
Il parametro **group** permette di raggruppare una sequenza di azioni con una etichetta che è il valore passato in questo campo, se passato *blank non saranno visibili raggruppamenti di opzioni.
Il parametro **last** indica che l’opzione passata è l’ultima e permette al componente di chiudere le opzioni.

Limiti della *COM:

il campo di ritorno non può essere un campo del subfile

se il componente è interno alla pagina questa può subire deformazioni se il programmatore non lascia spazio sufficiente

La funzione *COM **non è** compatibile con le WINDOW, quindi si possono verificare visualizzazioni non corrette nel caso in cui una finestra venga emessa sopra un formato contenente delle funzioni *COM. In generale le funzioni *COM si abilitano solo nella finestra attiva.

Per una corretta valutazione degli spazi di occupazione dei componenti e dello spazio a disposizione dell'area applicativa è sconsigliato l'uso del CSS variabile a favore di un unico CSS fisso.

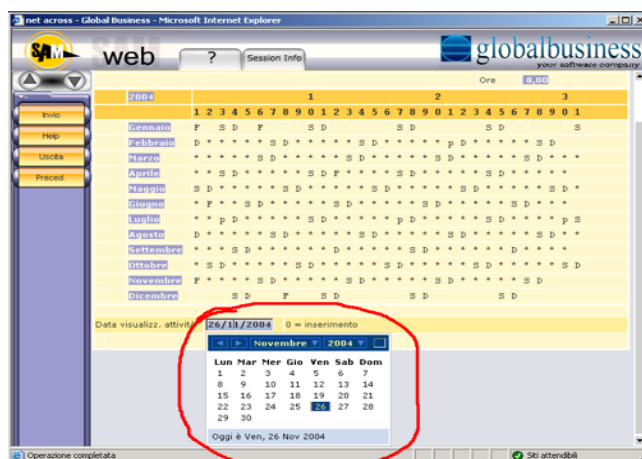
L'uso della tabella SQL (TABLE/SEARCH) deve essere moderato relativamente al numero di record estratti (nell'ordine max del migliaio di record) pena un eccessivo carico macchina probabilmente inutile.

*DAT

Questa keyword a livello di campo permette di abilitare su un campo data la funzione di ricerca tramite calendario. Al passaggio del mouse il puntatore evidenzia la possibilità di ricerca ed è possibile quindi richiamarla.
Come parametri prevede:

f:<formato data>	(se omesso gg/mm/aa) (inserire combinazione gg mm aa o aaaa con il separatore (/, -, ecc.))

Esempio:
se il campo prevede l'inserimento di una data indicare nel campo testo dello stesso:

```
NA*DAT f:gg/mm/aaaa
```

NetA.viewCalendarFor(nome_campo,formato)
E' possibile richiamare la visualizzazione del calendario grazie alle funzioni javascript , richiamando la funzione su un'immagine clikkabile. La funzione riceve il nome del campo destinatario del valore di ritorno e il formato della data selezionata.

Esempio:

```
A*NA*IMG V:"/images/calendar.png" P:N
A*   *EVT t:onclick j:$NetA.viewCalendarFor("TSTDTA2","gg/mm/aa");
A*   *CLS v:mycalendar
```

Al momento della selezione di una data e quindi al cambio del valore nel campo di ritorno viene generato un evento (onchange) sul campo di input stesso, come se la data fosse stata inserita manualmente.

*DEL

Questa keyword a livello di campo permette di escludere il campo dalla pagina HTML. Tale keyword è utile soprattutto quando a livello di DDS sono presenti decodifiche di campi che vengono trasferite a livello di menù a discesa oppure nelle opzioni del subfile.
La keyword non ha parametri.
Se utilizzata su campi di input questi vengono

tolti dall'html visibile ma trasformati in campi hidden, disponibili a ricevere valori tramite javascript.

Esempio:

per cancellare un campo indicare nel campo testo dello stesso:

```
NA*DEL
```

*EHT

Questa keyword a livello di record permette di creare un bottone per richiamare un editor HTML integrato in NetAcross. Tale keyword è utile quando si vuole creare o modificare un testo in formato HTML da inoltrare a funzioni particolari quali ad esempio invio di e-Mail o documenti con contenuto graficamente evoluto.
L'editor integrato infatti permette di editare il testo inserito in un contesto da editor di documenti.
Come parametri prevede:

d:<testo> (testo del pulsante che viene creato)

Il pulsante utilizza la classe ***btneditor*** che può essere personalizzata nel CSS.
Per riempire e scaricare il testo HTML sono disponibili due programmi che devono quindi essere richiamati dal programma gestionale:

Per leggere il file html o porzione di esso, e posizionarlo nell'area temporanea dell'editor:

```
Call      'NAEHTUPL'
Parm                     p_PathApp      256
Parm                     p_err            1
```

Per scaricare l'area temporanea in un file html o porzione di esso (indicato dal percorso).

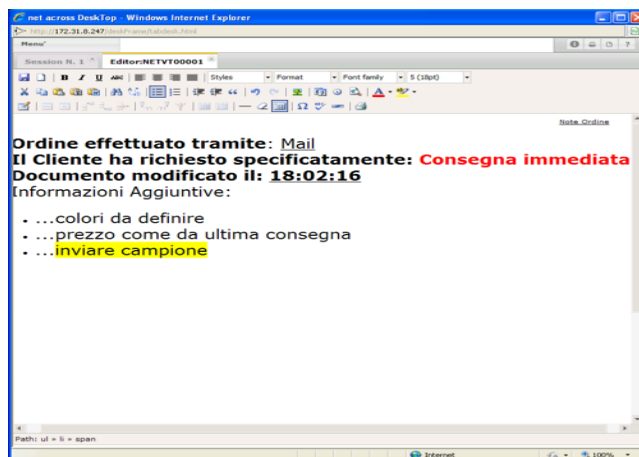
```
Call      'NAEHTDWN'
Parm                     p_PathApp      256
Parm                     p_err            1
```

Il Path è un percorso assoluto ad un file di testo contenente html.
Il codice di errore può essere:
A = Errore in reperimento dati di sessione
B = Errore in OpenFile() File appoggio
C = Errore in OpenFile() File Origne
D = Errore in Read File Origine
E = Errore in Write File Appoggio
F = Errore in attributi/autorizzazioni file origine

Al momento la versione di tinyMCE presente è la 3.4.1 ma periodicamente verrà implementata una versione più recente in base alle compatibilità dei browser e del prodotto.

Esempio:

se si vuole aggiungere un pulsante di gestione note HTML alla pagina note inserire nella sezione record dello stesso:

NA*EHT d:"Note HTML"

*EVT

Questa keyword a livello di campo permette di gestire gli eventi di un campo. Tale keyword è utile quando si vogliono automatizzare delle funzioni, interagire con le azioni dell'utente o creare "effetti speciali". La keyword prevede l'esecuzione di codice JavaScript e vengono fornite funzioni che prevedono alcune funzioni base per il trattamento dei dati.
Saranno disponibili gli eventi gestiti dai browser sui tipi di elementi html selezionati. Gli eventi gestibili sono per esempio onmouseover, onclick. Il JavaScript potrà usufruire di dati provenienti anche da altri campi. Nel parametro v: si possono intercalare delle costanti racchiuse fra doppi apici.
Come parametri prevede:

t:<tipo di evento>	(obbligatorio e ripetibile n volte) (separare gli eventi con uno spazio)
j:§<codice JavaScript>	(obbligatorio e ripetibile n volte) (ci deve essere uno script per evento) (ogni script deve iniziare con **§**)
v:[<campo>](<lunghezza>)	(facoltativo e ripetibile n volte)

All'interno del codice JavaScript si possono fornire questi valori speciali:

*value	Valore del parametro v:
name:<nome campo>	Risolve un nome campo all'interno di un subfile

Gli eventi specificabili nel parametro t: sono quelli standard previsti dal JavaScript e dipendono dall'oggetto su cui vengono inseriti. Segue un elenco dei più comuni:

onblur	perso il focus
onchange	cambiato il valore

onclick	click
ondblclick	doppio click
onfocus	ottenuto il focus
onmouseover	il puntatore del mouse è sopra il campo
onmousedown	il tasto del mouse è premuto sopra il campo
onmouseout	il puntatore del mouse è uscito dal campo
onkeydown	premuto tasto
onkeypress	premuto tasto e rilasciato
onkeyup	rilasciato tasto

I campi a cui si può far riferimento vengono riempiti di zeri a sinistra se numerici ed eliminati di spazi a destra se alfabetici.
Limitazioni: solo nella modalità interattiva e all'interno di un subfile il campo riferito dal parametro v: deve risiedere sulla stessa riga.

Esempio:
se abbiamo un campo codice articolo e codice colore e si vuole far apparire quando si passa sopra col mouse l'immagine in formato Jpeg corrispondente residente nella directory Articoli indicare nel campo testo dei campi stessi (*value vuol dire la concatenazione del valore dei campi):

```
NA*EVT t:onmouseover j:$NetA.get("/Articoli/"+*value+".jpg") v:[CODART](5)[COLORE](6)
```

*EXT

Questa keyword a livello di campo permette di eleggere il campo a campo speciale. Un campo speciale è un campo sul quale il cursore assume la forma di un dito e selezionando il tasto destro presenta una serie di funzioni speciali definite per quel campo. L'elenco delle funzioni corrispondenti ad una funzione estesa e le azioni ad esse corrispondenti vengono inserite con un apposito programma definito nel paragrafo Gestione Funzioni Estese in questo manuale. Tramite questo programma sarà possibile anche limitare gli accessi alle funzioni speciali in funzione di regole di autorizzazione.
Per fare un esempio si può abilitare una funzione estesa sul codice cliente che richiama le funzioni di estratto conto, elenco ordini, elenco documenti, ecc. Questa funzione permette quindi di espandere la funzionalità del programma originale consentendo di accedere a delle funzioni generalizzate e focalizzate sull'oggetto (ad esempio il cliente).
La keyword prevede di passare al gestore della funzione estesa uno o più parametri che sono in sostanza i dati dell'oggetto a cui si fa riferimento. Ad esempio per richiamare la funzione estesa Cliente/Fornitore dovranno essere forniti come parametro il tipo cliente/fornitore ed il codice cliente/fornitore. La composizione del richiamo potrà usufruire quindi di dati provenienti anche da altri campi.
Come parametri prevede:

v:<oggetto estensione>(<parametro>,...) (obbligatorio)

Oggetto estensione: è il nome dell'estensione inserito con Gestione Funzioni Estese.
Parametro: sono i parametri da passare eventualmente alla funzione estesa. Il numero di

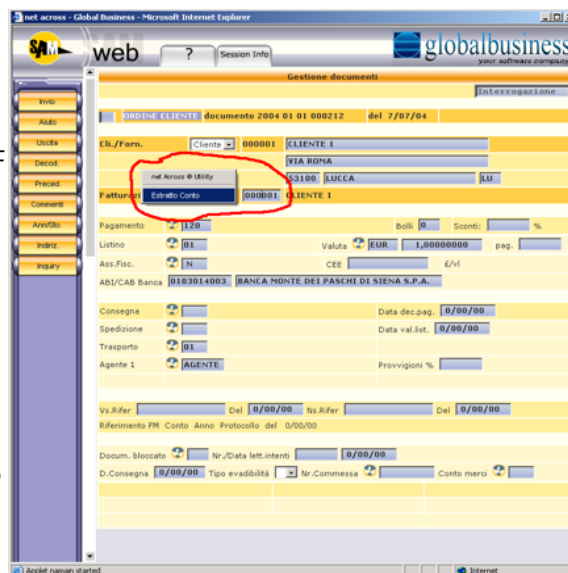
parametri dipende dall'implementazione di ogni funzione. Per recuperare il valore del campo in cui e attivata la funzione si indica il parametro speciale *value. Per inserire una costante si inserisce la stessa fra apici singoli. Per riferirsi ad un altro campo si indica il nome fra parentesi quadre, seguito dalla lunghezza fra parentesi tonde senza spazi fra i due. Se non viene indicata la lunghezza viene assunta la lunghezza del campo.

Esempio:

se si vuole attivare la funzione speciale Cliente sul codice cliente e passare alla funzione i campi TIPCF e se stesso inserire nel campo testo del codice cliente:

```
NA*EXT v:CLIENTE([TIPCF](1), *value)
```

Nella modalità *deskFrame* , è possibile fare in modo che le funzioni associate al campo siano visibili nella barra delle funzioni in alto a destra, la barra scorre se il testo contenuto è più esteso dello spazio disponibile. Per attivare la modalità è necessaria una sintassi leggermente diversa nella definizione della funzione estesa, nel campo descrizione invece del testo lineare, deve essere espresso così:

```
Descrizione :       [testo_descrittivo]{path_to_image}
```

*GAU

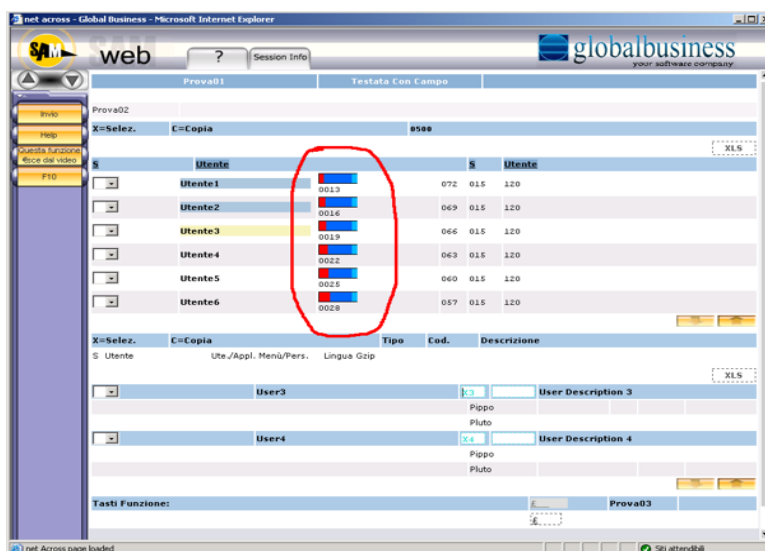
Questa keyword a livello di campo permette di trasformare un campo in un gauge (misuratore) grafico. Tale oggetto grafico è utile quando si vogliono rappresentare dei valori numerici o dei rapporti fra loro in modo evidente ed intuitivo. La keyword prevede di:

1. Rappresentare il valore di un singolo campo usando due colori.
2. Rappresentare il valore di un singolo campo usando diversi colori in funzione del raggiungimento di valori percentuali.
3. Rappresentare il rapporto percentuale del valore di più campi usando un colore per ogni campo.

Come parametri prevede:

s:<ampiezza campo>	(Ampiezza in pixel del campo misuratore. Default 100)
c:<colore di foreground>	(Colore di primo piano espresso nella forma #RRGGBB R=Rosso G=Green B=Blue. Default #FF0000 (Rosso))
b:<colore di background>	(Colore di sfondo espresso nella forma #RRGGBB R=Rosso G=Green B=Blue. Default #FFFFFF (Bianco))
d:<Y/N>	(Cancellare il valore numerico Y=Rappresenta solo il misuratore.

	N=Rappresenta valore numerico e misuratore. Default=Y
r:<lim.infer.> l<lim.super.> [<n.campo>] [<n.campo>]	Ampiezza minima e massima di valori rappresentabili dal misuratore in valore assoluto (numero) o relativo (nome campo non di subfile). Non significativo se espresso il parametro n:. Default 0l100
p:[<perc>](colore)	Assegnazione delle percentuali di riempimento ai colori. Il colore deve essere nella forma come il colore di foreground. Le coppie all'interno del parametro p: sono ripetibili fino a 7 volte. Esclusiva col parametro n:.
n:[<n.campo>](colore)	Assegnazione dei nomi dei campi di riempimento ai colori. Il colore deve essere nella forma come il colore di foreground. Le coppie all'interno del parametro n: sono ripetibili fino a 7 volte. Esclusiva col parametro p:.

Limitazione: solo nella modalità interattiva e all'interno di un subfile se si fa riferimento ad un campo questo deve risiedere sulla stessa riga.

Esempi:

1. Rappresentare il valore di un singolo campo usando due colori.
   Se abbiamo un campo che rappresenta un valore di misurazione nel limite 0-100 e lo vogliamo trasformare in visualizzazione grafica indicare nel campo testo dello stesso:
   NA*GAU
2. Rappresentare il valore di un singolo campo usando diversi colori in funzione del

raggiungimento di valori percentuali.
Se abbiamo un campo che rappresenta un valore di misurazione e lo vogliamo trasformare in visualizzazione grafica multicolore indicare nel campo testo dello stesso:
NA*GAU s:50 d:N p:[33](#0066FF)[66](#00BBFF)[100](#00FFFF)

3. Rappresentare il rapporto percentuale del valore di più campi usando un colore per ogni campo.
   Se abbiamo due campi che rappresentano un rapporto e lo vogliamo trasformare in visualizzazione grafica multicolore indicare nel campo testo di uno dei due campi o di un campo creato come segnaposto:
   NA*GAU s:50 d:N n:[V1NUM2](#0066FF)[V1NUM3](#00BBFF)

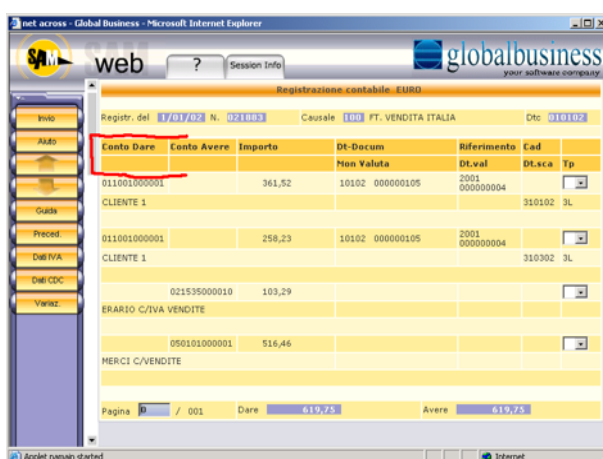
---

*HDR

Questa keyword indica in presenza di un subfile di quante righe salire per creare un'unica tabella fra le righe stesse e la testata testuale. Se non viene indicata la keyword viene compresa una riga sopra il subfile.
La salita si ferma quando viene incontrato un campi di input oppure una riga vuota. Per forzare la salita in modo incondizionato basta aggiungere una F subito dopo il numero riga (ad es. 2F).
Come parametri prevede:

<numero riga>	(obbligatorio con valori da 2 a 5. Aggiungere una F per non fermarsi in presenza di campi di input o righe vuote)

Esempio:
per comprendere due righe di testata nella struttura di incolonnamento del subfile inserire nel campo testo del subfile record (non il control):
NA*HDR 2

---

*HLP

Questa keyword permette collegare ad una pagina NetAcross un testo di aiuto in formato multimediale. La keyword può essere espressa a livello di record ed in questo caso è richiamata tramite una richiesta generica di help. Se espressa a livello di campo punta ad una ancora

dell'Help e viene richiamata con una richiesta di aiuto effettuata sul campo stesso.
Le richieste di aiuto a livello di campo sono disponibili sono in modalità batch.
A livello di configurazione del server Http occorrerà aggiungere la pubblicazione della cartella di root dell'Help. Ad esempio:

```
Alias /helpdoc /help
# abilita le richieste per *HLP
<Directory /help/>
        Order allow,deny
        Allow from All
</Directory>
```

A livello di configurazione NetAcross (Config.xml) occorrerà aggiungere la keyword che collega la pubblicazione appena creata. Ad esempio:

```
<HelpRoot>/helpdoc</HelpRoot>
```

Riferimenti in configurazione

CONFIG.XML

<JScriptPers>*path_assoluto_javascript*</JScriptPers>: permette di indicare il percorso di un file aggiuntivo *.js contenente funzioni JavaScript personalizzate da aggiungere a tutte le pagine dinamiche

Come parametri prevede:

v:<percorso Help>	(facoltativo. Indica il percorso all'interno della root dell'help del file che contiene il testo di aiuto. Se non indicato sarà <nome file>.html)

Esempio:
per indicare il puntamento ad un testo di help per l'intero record inserire a livello di record come segue:
NA*HLP
Questo punterà ad un Html con lo stesso nome del formato record.
Oppure indicare:
NA*HLP v:gestioneordini.html
per puntare ad un Html specifico.
A livello di campo indicare:
NA*HLP v:'codcli'
Questo punterà all'ancora codcli all'interno dell'Html puntato dal formato video.

*HNA

Questa keyword indica il nome del campo hidden con cui proseguire la ricerca del testo per le keyword NetAcross nel caso non sia sufficiente lo spazio a disposizione (50 caratteri) del campo. Se non viene indicata la keyword viene assunto il nome del campo stesso.
Al nome risultante verrà comunque aggiunto il suffisso Hx dove x è un numero che va da 1 a 9.
La sintassi deve essere nella forma *HNA=<nome> dove <nome> è il nome del campo hidden su cui proseguire (non comprensivo del suffisso Hx).
Questa keyword è particolarmente utile nei file video utilizzati da programmi RPG OPM dove è obbligatoria una lunghezza massima di 6 caratteri e che il suffisso Hx può violare costringendo ad inserire specifiche Rename nella righe di Input.
Esempio:
per definire come campi di continuazione i campi hidden CODH1 e CODH2:
NA*HNA=COD

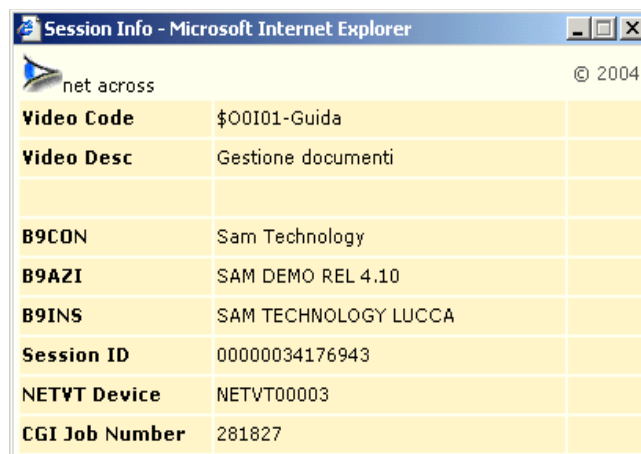
*INF

E' abitudine diffusa inserire su ogni pagina video alcuni campi generici: nome del programma, titolo, ditta, ecc. Questi campi occupano costantemente un'area del video e tolgono quindi spazio ai dati. NetAcross prevede la possibilità di nascondere questi campi e farli vedere solo a richiesta, liberando così lo spazio per altri campi della pagina.
Questa keyword a livello di campo permette di indicare al generatore di Html che la costante o il campo di output deve essere estratta dalla pagina e portata nella visualizzazione a richiesta. E' possibile estrarre un campo al solo fine di renderlo facilmente disponibile per le funzioni JavaScript. E' possibile decidere se cancellare o meno il campo dalla pagina originale.
A JavaScript il valore della variabile sarà disponibile con la sintassi NetA.<id campo>.value
Per la pagina di visualizzazione a richiesta sono previsti una serie di campi standard (sulla base dell'id campo):

VideoCode	Codice del programma/file video
VideoDesc	Descrizione del programma (Titolo)
VideoDesc2	Descrizione supplementare(Funzione)
Info1 o INFO1	Es: Versione Prodotto
Info2 o INFO2	Es: Ditta
Info3 o INFO3	Es: Installazione

se l'id campo non è nella lista questo viene esposto al JavaScript ma non inserito nella visualizzazione a richiesta.

Come parametri prevede:

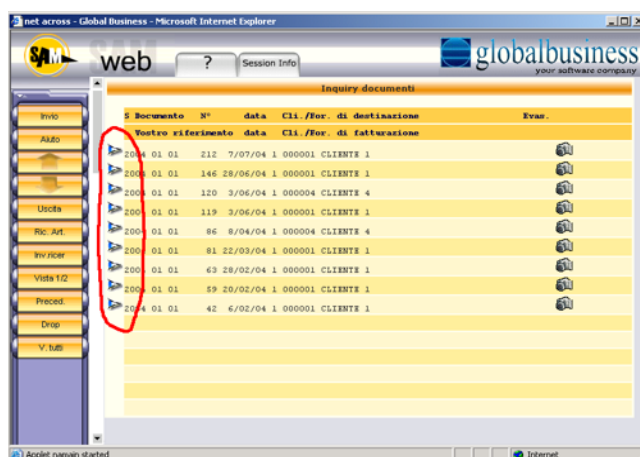
i:<id campo>	(obbligatorio. Viene assegnato un nome di campo che sarà facilmente visibile nella pagina Html, ad esempio da un JavaScript.)
s:<ampiezza>	(facoltativo. E' l'ampiezza in caratteri del campo. Se non inserito vale la lunghezza del campo.)
d:<Y/N>	(facoltativo. Se cancellare il campo dalla pagine originale. Se non inserito vale N.)

Esempio:
se un campo rappresenta il nome del programma indicare nel campo testo dello stesso:
NA*INF i:VideoCode d:Y
Esempio:
se un campo rappresenta il titolo della pagina indicare nel campo testo dello stesso:
NA*INF i:VideoDesc

Session Info - Microsoft Internet Explorer

net across © 2004

Video Code	$00I01-Guida
Video Desc	Gestione documenti
B9CON	Sam Technology
B9AZI	SAM DEMO REL 4.10
B9INS	SAM TECHNOLOGY LUCCA
Session ID	00000034176943
NETVT Device	NETVT00003
CGI Job Number	281827

*IMB

Questa keyword a livello di campo permette di trasformare un campo in una o più immagini sensibili al click. Tale keyword è utile quando il campo prevede uno o più valori e quindi è più semplice selezionarli con un click. In particolare questa keyword è indicata nel campo scelta dei subfile a scelta singola.
Come parametri prevede:

v:<valore da inserire nel campo>	(obbligatorio)
i:<path dell'immagine da usare>	(se omesso mette immagine di default)
s:<ampiezza immagini>	(Ampiezza in pixel delle immagini. Facoltativo.)

Possono essere inseriti più valori (valore e immagine), separati da punto e virgola. L'ampiezza immagini deve essere inserita una sola volta ed è utile quando se ne prevedono più di una per evitare che vengano disposte in senso verticale.

Esempio:
se un subfile prevede come unica scelta una X per selezionare una riga e si vuole l'invio automatico cliccando sull'immagine eye.gif indicare nel campo testo del campo scelta:
NA*IMB v:X i:/images/eye.gif

Se si vogliono inserire più opzioni (X=scelta e C=cancella) la sintassi è:
NA*IMB v:X;C i:/images/eye.gif;/images/canc.gif

Un altro modo per creare un bottone è definire un campo di **input/output** con l'attributo scambio tastiera di tipo **I** (inserimento tramite mouse). Il campo verrà trasformato in un bottone cliccabile.
<input type="button" />

Il contenuto del bottone sarà il valore del campo stesso. A livello di 5250 il campo sarà utilizzabile solo con il mouse.

*IMG

Questa keyword a livello di campo permette di trasformare un campo in un immagine. Tale immagine è utile quando il campo può essere referenziato ad un'immagine, ad un documento, ecc. Può essere usata anche in una lista (subfile). All'immagine sarà possibile anche associare un evento da gestire tramite la keyword EVT.
Se si vuole visualizzare un'immagine intermedia (ad esempio una macchina fotografica) per poi cliccarci sopra per ottenere l'immagine effettiva usare la keyword IML.
Visualizzare le immagini direttamente (soprattutto in una lista) implica il trasferimento al browser immediato dell'immagine effettiva e può comportare un elevato trasferimento di dati verso il browser.
La composizione del richiamo potrà usufruire di dati provenienti anche da altri campi. Se indicato il campo di riferimento da tale campo verrano utilizzate le modalità di visibilità: se tale campo non viene visualizzato scompare anche l'immagine corrispondente. Nel parametro v: si possono intercalare delle costanti racchiuse fra doppi apici.
Come parametri prevede:

v:[<campo>](<lunghezza>)	(obbligatorio e ripetibile n volte)
r:<campo di riferimento>	(se omesso vale se stesso)
w:<n.pixel di larghezza>	(se omesso vale la dimensione originale)
h:<n.pixel di altezza>	(se omesso vale la dimensione originale)
p:<Y/N>	(definisce se per ottenere l'immagine si punta direttamente alla directory oppure si passa attraverso alias protetto). Y=Server documenti N=Accesso diretto a directory (default))

I campi a cui si può far riferimento vengono riempiti di zeri a sinistra se numerici ed eliminati di spazi a destra se alfabetici.
Limitazione: solo nella modalità interattiva e all'interno di un subfile il campo riferito dal parametro v: deve risiedere sulla stessa riga. All'interno di un subfile la keyword fa riferimento al record del subfile senza bisogno della sintassi *name.

Esempio:
se abbiamo due campi codice articolo e colore e si vuole creare un'immagine relativa ai campi il cui valore corrisponde ad una immagine residente nella directory Articoli indicare nel campo testo di un campo appositamente aggiunto per diventare immagine (*value vuol dire la concatenazione del valore dei campi):

`NA*IMG v:"/Articoli/"[CODART](5)[COLORE](6)".jpg")`
Se specificato accesso tramite server documenti (parametro p:) deve essere configurato l'Alias corrispondente per ricevere la funzione (vedere Gestione Alias/Directory in questo manuale).

*IML

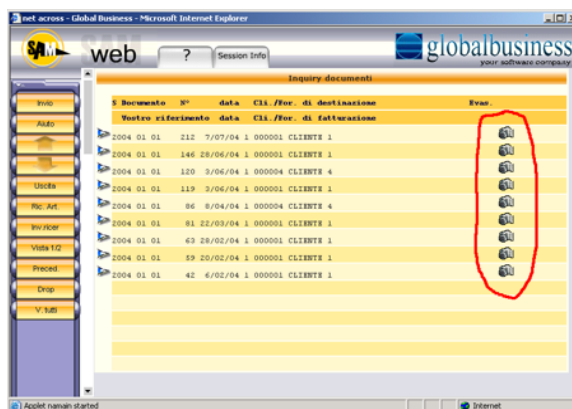
Questa keyword a livello di campo permette di trasformare un campo in un immagine sensibile al click. Tale immagine è utile quando il campo può essere referenziato ad un'immagine, ad un documento, ecc. La keyword prevede l'esecuzione di codice JavaScript e vengono fornite funzioni che prevedono le funzioni base di reperimento dati dall'immagine. Può essere usata anche in una lista (subfile).
L'immagine in cui viene trasformato il campo è un'immagine standard (ad esempio una macchina fotografica) su cui bisogna cliccare per ottenere l'immagine effettiva richiesta. Questa tecnica è veloce e non implica il trasferimento al browser immediato dell'immagine effettiva. Se si vuole visualizzare l'immagine direttamente senza bisogno di cliccarci sopra usare la keyword IMG.
La composizione del richiamo potrà usufruire di dati provenienti anche da altri campi. Se indicato il campo di riferimento da tale campo verrano utilizzate le modalità di visibilità: se tale campo non viene visualizzato scompare anche l'immagine corrispondente. Nel parametro v: si possono intercalare delle costanti racchiuse fra doppi apici.
Come parametri prevede:

v:[<campo>](<lunghezza>)	(obbligatorio e ripetibile n volte)
r:<campo di riferimento>	(se omesso vale se stesso)
i:<immagine da visual. come link>	(se omesso vale /images/foto.gif)
j:<codice JavaScript>	(obbligatorio)

All'interno del codice JavaScript si possono fornire questi valori speciali:

*value	Valore del parametro v:
name:<nome campo>	Risolve un nome campo all'interno di un subfile

I campi a cui si può far riferimento vengono riempiti di zeri a sinistra se numerici ed eliminati di spazi a destra se alfabetici.
Limitazione: solo nella modalità interattiva e all'interno di un subfile il campo riferito dal parametro v: deve risiedere sulla stessa riga.

Esempio:
se abbiamo due campi anno fattura e numero fattura e si vuole creare un'icona cliccabile relativa ai campi il cui valore corrisponde ad una documento di tipo PDF rappresentante la fattura in formato grafico residente nella directory Fatture indicare nel campo testo di un campo appositamente aggiunto per diventare immagine

(*value vuol dire la concatenazione del valore dei campi):
`NA*IML v:[ANNOFT](4)[NUMFT](6) j:NetA.get("/Fatture/"+*value+".pdf")`
Se si utilizza la funzione JavaScript NetAcross NetA.get deve essere configurato l'Alias corrispondente per ricevere la funzione (vedere Gestione Alias/Directory in questo manuale).

---

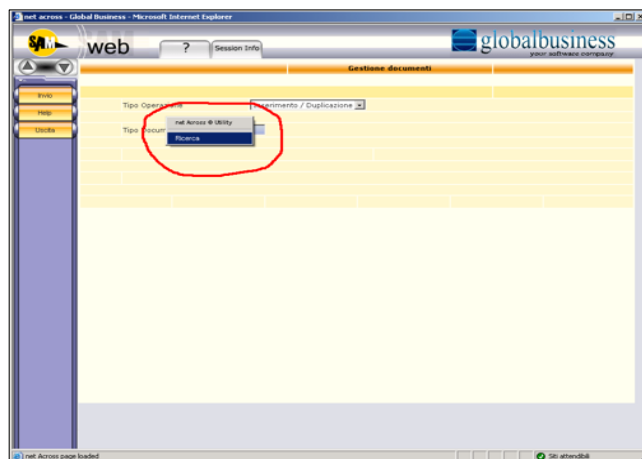
*INQ

Questa keyword a livello di campo permette di abilitare un campo alla funzione di ricerca (prevista dal programma AS400). Al passaggio del mouse il puntatore evidenzia la possibilità di ricerca ed è possibile quindi richiamarla.
Come parametri prevede:

v:<valore da inserire nel campo> (se omesso vale ?)

E' possibile modificare il comportamento standard della keyword creando una funzione estesa col nome standard INQUIRY (vedere Gestione Funzioni Estese in questo manuale).

- Se tale oggetto non esiste viene eseguito il JavaScript standard *NetA.moveOption('%1','%2')* dove *%1* è il nome del campo su cui è impostata la keyword e *%2* è il parametro v: .
- Se tale oggetto esiste viene eseguito il codice associato a tale funzione (normalmente un JavaScript). In questo modo è possibile gestire ad esempio la ricerca tramite un tasto di comando (usando ad esempio la funzione NetA.F('F4')).

Esempio:
se il campo cliente prevede di inserire ? per effettuare la ricerca clienti indicare nel campo testo dello stesso:
`NA*INQ`

***JCH**

Javascript Charts, permette di generare grafici in HTML5 e SVG (rende deprecata la *CHR).

Come parametri prevede:

t:<tipo grafico>	(unico valore valido è "STATIC" per default)
i:<ID univoco del grafico>	(ID univoco all'interno della pagina per identificare il grafico)
w:<larghezza>	(Larghezza in pixel del grafico)
h:<altezza>	(Altezza in pixel del grafico)
r:<campo di riferimento>	(campo di riferimento)
d:<titolo del grafico>	(se presente viene usato per titolo del grafico e può contenere un testo o un campo non di subfile)

I parametri **r:** e **d:** nono sono obbligatori, tutti gli altri si.
La keyword è a livello di campo e sostituisce il grafico al campo, per default il grafico "si fa spazio" nella videata, ma è possibile cambiarne il comportamento tramite fogli di stile.

A runtime viene cercato il file di definizione del grafico, il file di definizione è un file javascript e deve essere nominato secondo questa regola:

*/document_root/*CHART/*Nome_Applicazione*/BASE/ITA/*Nome_Libreria/Nome_FileVideo.Nome_FormatoExmft.ID_univico_grafico.js*

document_root: di solito è la cartella NETACROSS
Nome_Applicazione: come definito sul profilo utente di validation list
Nome_Libreria: è la cartella con lo stesso nome della libreria che contiene l'oggetto DSPF

Il nome del file è composto dal nome del file video "." (punto) il nome del formato di exfmt più l'ID identificativo del grafico come definito sul sorgente dal parametro **i:**

Questo permette di inserire nello stesso html più grafici (anche su formati diversi che compongono la videata)
Il file di definizione dei grafici può essere anche un solo file javascript ma per comodità di manutenzione e di logica è possibile usare un file di definizione per ogni grafico

Quindi un file javascript contiene le specifiche di definizione del grafico.
I grafici sono prodotti con la libreria **C3** (http://c3js.org/) basato a sua volta sulle librerie D3 (http://d3js.org/) che sono estremamente potenti e versatili.
Di fatto l'utente può utilizzare i suoi plugin sviluppati per D3 all'interno di netacross, se ha necessità di integrare le pagine con elementi grafici di tipo data-driven.

La sintassi di definizione e le possibilità parametriche dei grafici sono quelli di C3 ma sono integrate con la logica dei template di netacross.
In pratica ogni campo della videata (non subfile) è utilizzabile all'interno del file di configurazione

se indicato dalla sintassi %NOMECAMPO%.
L'uso dei campi del subfile al momento è limitato alla presenza di un subfile esteso.

Esempio: grafico basato su campi del video

Il grafico è identificato dall'id ID=CHART1 e viene "agganciato" con la sintassi bindto: '#ID' che sarà un default di tutte le definizioni.
Il campo %V1CHT% è un campo di testata e viene usato nella prima riga dei dati del grafico come intestazione dello stesso.
I campo %V1CHN1% e gli altri sono del subfile e la logica del template permette di replicare la riga per i record presenti nel subfile al momento dell'EXFMT

```
var chart1 = c3.generate({
  bindto: '#CHART1',
  data: {
          rows: [
            ['%V1CHTI%', 'data2', 'data3'],
            [%V1CHN1%, %V1CHN2%, %V1CHN3%],
          ],
          unload: true,
          onclick: function (d, element) { console.log("onclick", d, element); },
          onmouseover: function (d) { console.log("onmouseover", d); },
          onmouseout: function (d) { console.log("onmouseout", d); },
        }
      });
```

Il file viene processato e poi pubblicato, in modo che venga scaricato dalla pagina html e attivato per disegnare il grafico.

Per poter visualizzare dei dati non direttamente presenti sul video è possibile utilizzare un sintassi diversa che permetta di caricare un documento in formato .CSV (comma separated values) contenente i vettori dei dati dei grafici.
Il file .csv deve essere disponibile al download al momento della EXFMT quindi deve essere residente sul file system o generato precedentemente.
Nell'esempio il file dei dati è posizionato nella cartella /netacross/chart/temp/con il nome del job (recuperato con la variabile di sessione NetA.netvt) + l'identificativo del grafico + una data generata al volo per evitare la cache di alcuni browser.

Esempio: Grafico con dati esterni

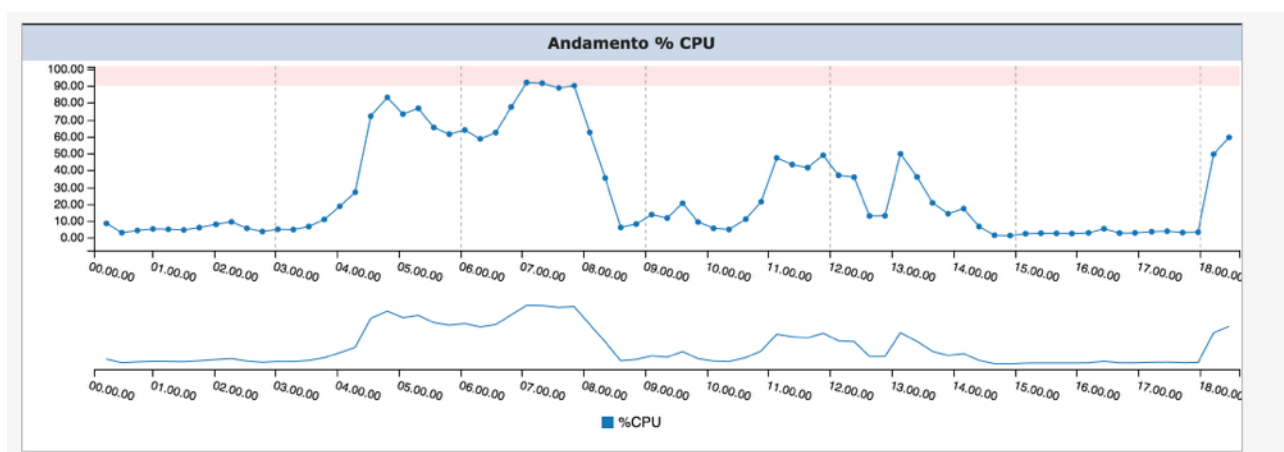
```
var d = new Date();
var inizio = "%VVDATAA%";
var ticks = new Date(inizio).getTicks(60);
var chart1 = c3.generate({
            bindto: '#CHART1',
            data: {
          x: 'X',
          url: '/chart/temp/' + NetA.netvt + '_CHART1.csv'+'?_='+d,
          xFormat: '%Y-%m-%d-%H.%M.%S'
        },
         axis: {
                y: {
                  tick: {
                  format: d3.format("3.2f")
                  }
                },
```

```
            x: {
              type: 'timeseries',
              tick: {
                values: ticks,
                format: "%H.%M.%S",
                rotate: 15,
              }
            }
          },
        grid: {
                x: {
                  show: true
                }
              },
        zoom: {
                enabled: true,
                rescale: true,
                onzoomstart: function (event) {
                  console.log("onzoomstart", event);
                },
                onzoomend: function (domain) {
                  console.log("onzoomend", domain);
                },
              },
              subchart: { show: true },

        regions: [{axis: 'y', start: 90, class: 'regionYRed'},]

    });
```

Il contenuto del file di dati è questo :

```
"X","%CPU"
2015-06-10-18.28.07,59.40
2015-06-10-18.12.57,49.50
2015-06-10-17.57.45,3.40
2015-06-10-17.42.33,3.20
2015-06-10-17.27.21,4.00
2015-06-10-17.12.09,3.70
2015-06-10-16.56.58,3.00
2015-06-10-16.41.46,2.90
2015-06-10-16.26.34,5.40
2015-06-10-16.10.47,3.00
2015-06-10-15.55.36,2.60
2015-06-10-15.40.25,2.70
2015-06-10-15.25.13,2.80
2015-06-10-15.10.01,2.50
2015-06-10-14.54.51,1.40
2015-06-10-14.39.41,1.60
2015-06-10-14.24.30,6.70
2015-06-10-14.09.20,17.30
2015-06-10-13.54.10,14.30
2015-06-10-13.39.00,20.70
…
```

dove il primo campo è un orario e il secondo è un valore percentuale con 2 decimali.

Il risultato della definizione con i dati scaricati è più o meno questo

Le opzioni disponibile nella librearia C3 sono molte e in continua evoluzione, ogni qual volta ci saranno evoluzioni significative nella libraria questa verrà aggiornata nelle prime PTF di prodotto disponibili.

I grafici sono anche auto alienabili per cui è possibile aggiornarli periodicamente e sono sensibili agli eventi del mouse o touch screen e quindi possono pilotare delle funzionalità o permettere la navigazione grafica di informazioni.

Per un esempio completo delle tipologie di grafici disponibili si rimanda all'idirizzo:
http://c3js.org/examples.html

***JSC**

Questa keyword a livello di record indica il nome di un file esterno contenente codice JavaScript da associare alla pagina Html attuale. La path è relativa alla Document Root del Server Http e deve essere pubblicata all'esterno. Può essere utilizzata la directory Scripts fornita con NetAcross che contiene le funzioni base del prodotto.
Il nome va inserito come parametro della keyword con o senza doppi apici di delimitazione e con la barra iniziale.
In questo file JavaScript possono essere inserite le funzioni definite dall'utente che possono essere poi richiamate dalle altre keyword nella pagina.

E' disponibile una funzione speciale che se definita viene richiamata dall'evento di visualizzazione della pagina associata:

```
function initpage(){
 ....
}
```

questa funzione se definita nel file javascript utente, viene richiamata all'avvio della pagina prima che l'utente possa eseguire alcuna operazione e permette personalizzare il funzionamento e la presentazione della pagina per esempio inserendo codice o creando gestori di eventi ad hoc.

Esempio:
per aggiungere il file ScriptPers.js residente nella directory /NetAcross/Scripts e renderne disponibili le funzioni nella pagina inserire nel campo testo di un record contenuto nella pagina (il nome dell'istanza, in questo caso NetAcross, non va indicato):
`NA*JSC /Scripts/ScriptPers.js`

---

*LNK

Questa keyword a livello di campo permette di trasformare un campo in un link. Tale link è utile quando il campo può essere referenziato ad un'immagine, ad un documento, ecc. La keyword prevede l'esecuzione di codice JavaScript e vengono fornite funzioni che prevedono le funzioni base di reperimento dati dal link. La composizione del link potrà usufruire di dati provenienti solo da questo campo.
Come parametri prevede:

j:<codice JavaScript>	(alternativo ad h:)
h:"testo_fisso" oppure "*value"	(alternativo a j:)
l:"*label_descrittiva*"	(opzionale)

Il parametro **h:** permette di impostare un href come testo fisso o con il valore speciale "*value" recuperando l'indirizzo dal testo del campo.

Per **j:** all'interno del codice JavaScript si possono fornire questi valori speciali:

*value	Valore del campo senza spazi a destra contenente la keyword
name:<nome campo>	Risolve un nome campo all'interno di un subfile

I campi a cui si può far riferimento vengono riempiti di zeri a sinistra se numerici ed eliminati di spazi a destra se alfabetici.
Limitazione: solo nella modalità interattiva e all'interno di un subfile il campo riferito con <nome campo> deve risiedere sulla stessa riga.

Questa keyword può essere utilizzata anche per automatizzare operazioni. Ad esempio un elenco di opzioni in output e un campo scelta che richiede la digitazione del numero dell'opzione può essere automatizzato trasformando le varie opzioni in link che scrivono in valore dell'opzione nel campo scelta e premono invio. Il campo scelta viene tolto dal video con la keyword *DEL.

Esempio:
se abbiamo un campo articolo in cui il codice corrisponde ad una immagine di tipo Jpeg residente nella directory Articoli indicare nel campo testo dello stesso (*value vuol dire il valore del campo):
`NA*LNK j:NetA.get("/Articoli/"+*value+".jpg")`
Se si utilizza la funzione JavaScript NetAcross NetA.get deve essere configurato l'Alias corrispondente per ricevere la funzione (vedere Gestione Alias/Directory in questo manuale).

---

*MNU

Questa keyword a livello di campo permette determinare con che tipo di menù rappresentare un campo che prevede più valori espressi attraverso la keyword DDS Values. Il campo deve quindi avere a livello di DDS la keyword Values con l'elenco dei valori ammessi. Il parametro della keyword permette di scegliere il menù da attribuire fra Menù a discesa, Radio Button verticale o orizzontale, combo box e check box.
Se non viene inserita questa keyword i default di creazione sono:
Se le values sono due ed una delle due rappresenta uno spazio viene creato un check box.
Se le values sono due e nessuna delle due rappresenta uno spazio viene creato un radio button verticale.
Se le values sono più di due viene creato un select box.
Attenzione deve essere posta nella forzatura a check box, perché si può forzare questo controllo solo se la Values prevede due valori. Il primo viene considerato come True, il secondo come False. Si possono verificare incongruenze se la pagina viene emessa con un valore diverso da quelli permessi (ad esempio un campo che prevede S (true) e N (false) ma inizialmente viene emesso inizializzato a spazio.).
Come parametri prevede:

<valore>=<tipo menù>	(select = menù a discesa radioO = radio button orizzontale radioV = radio button verticale comboV = combo box con solo valore comboVD = combo box con valore e descrizione checkbox = check box)

Esempio:
se un campo ha la parola chiave Values con i valori S e N viene creato automaticamente un radio button con i valori indicati (perché si assume che in realtà i valori da prevedere siano S, N e spazio). Se si è certi che il campo viene inizializzato con un valore valido si può forzare a check box. Indicare nel campo testo dello stesso:
`NA*MNU v:checkbox`

Esempio:
se un campo viene rappresentato come radio button verticale (valore assunto se ci sono 2 valori

ma questi non sono conciliabili con la creazione di un check box) e se c'è spazio a sufficienza sulla destra può essere utile trasformarlo in un radio button orizzontale. Indicare nel campo testo dello stesso:
NA*MNU v:radioO

*MRG

Questa keyword a livello di record permette di riunire due record in una tabella HTML. Tale keyword è utile quando i record presentano dati collegati nel loro svolgimento, ad esempio una tabella in un record correlata ad un'altra in un altro record. Il generatore HTML incolonnerebbe i campi con una logica separata proponendo un'interpretazione poco leggibile. Con questa keyword è possibile indicare a quale riga DSPF iniziare una nuova logica di incolonnamento campi che dovrà essere collegata al record successivo.
E' possibile indicare una unica keyword per ogni composizione video.
Come parametro deve essere inserito il formato record a cui collegarsi ed il valore di riga alla quale iniziare una nuova tabella:

<nome formato>	(obbligatorio)
<numero riga>	(obbligatorio. Ammessi numero riga da 1 a 26)

Esempio:
per unificare la logica di incolonnamento di una pagina con un formato record con una prima parte logica compresa fra riga 1 e riga 5 ed una seconda parte compresa fra riga 6 e riga 10 che deve avere lo stesso incolonnamento del record FMT2 che occupa dalla riga 11 alla riga 24 indicare nel campo testo del record superiore:
NA*MRG FMT2 06

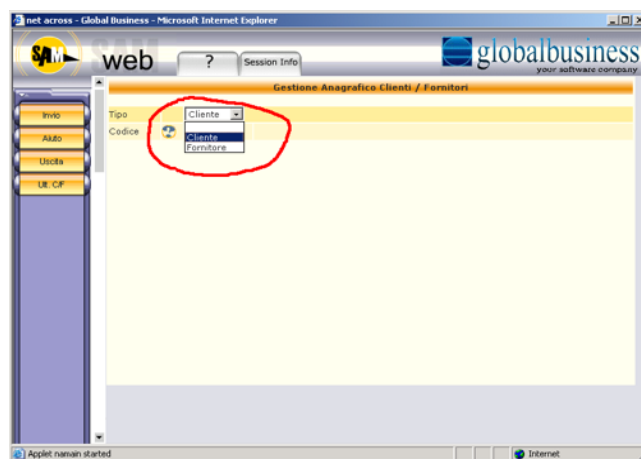
Se i formati record da unire comprendono un subfile valgono queste regole:
Se si vuole collegare un formato (es. FMT1) ad un formato subfile (es. FMT2) e quest'ultimo è inferiore a FMT1 si indica a livello di record in FMT1: NA*MRG FMT2 X. X rappresenta il numero di riga di FMT1 da cui iniziare l'unione.
Se si vuole collegare un formato (es. FMT3) ad un formato subfile (es. FMT2) e quest'ultimo è superiore a FMT3 si indica a livello di record in FMT3: NA*MRG FMT2 1. X rappresenta il numero di riga ed in questo caso vale sempre 1 (unisce tutto il record).

*SEL

Questa keyword a livello di campo (di input o input/output) permette di integrare ai valori le descrizioni di un campo con menù a scelta multipla. I parametri della keyword permettono di dare una descrizione ai valori di scelta. Il primo valore inserito è quello selezionabile con doppio click.
Si possono integrare le descrizioni sia fornendo la costante relativa oppure facendo riferimento ad un file messaggi residente su iSeries (per applicazioni multilingue).
E' disponibile anche la keyword *SEV che effettua le stesse funzioni di questa keyword partendo però da un elenco di valori variabile fornito da un programma di exit esterno.

Il numero massimo di valori gestibili è 40.
Come parametri prevede:

f:<file messaggi>	(nome del file messaggi su iSeries che contiene le descrizioni dei valori. Formato: *LIBL/<nome>, <libr>/<nome>, <nome> Facoltativo.)
<valore>=<descrizione>	(possono essere inseriti n per un max di 99 elementi valori separati da spazio. I valori devono corrispondere alla VALUES del campo ed avere la stessa lunghezza. Se è stato indicato il parametro f: nella descrizione si indica l'id del messaggio; altrimenti nella descrizione si inserisce la costante da associare al valore.)
s:<ampiezza campo>	(Ampiezza in caratteri del campo del menù. Facoltativo – default 32.)
m:<numero righe>	(Se inserito presenta il combo box aperto per il numero di righe specificate. Nel caso contrario il combo box è chiuso sull'opzione selezionata e può essere aperto con l'apposito tasto. Facoltativo.)

Esempio:
se un campo ha la parola chiave Values con i valori C e F viene creato automaticamente un menù di selezione con i valori indicati. Per rendere più leggibile il menù si può sostituire i valori con il loro significato C=Cliente F=Fornitore. Indicare nel campo testo dello stesso:
NA*SEL C=Cliente F=Fornitore
La stessa versione riferita a file messaggi (con traduzione quindi) va indicata così:
NA*SEL f:*LIBL/MSGAPPL C=USR0001 F=USR0002

*SEV

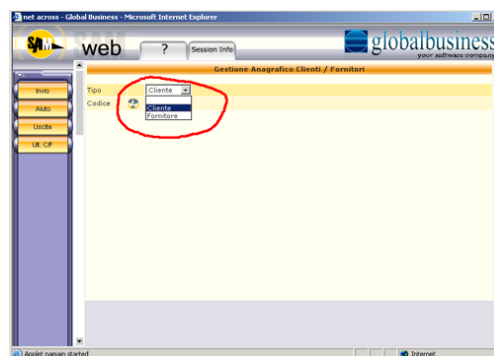
Questa keyword a livello di campo permette di definire un campo menù a scelta multipla e cui valori ammessi e le descrizioni relative vengono reperiti da un programma di exit esterno. Il primo valore ritornato dalla exit è quello selezionabile con doppio click.
E' disponibile anche la keyword *SEL che effettua le stesse funzioni di questa keyword partendo però da un elenco di valori e descrizioni fornito dalla keyword stessa.
Come parametri prevede:

v:[<campo>](<lunghezza>)	(Ripetibile n volte. Compone il valore passato al programma di exit. Facoltativo.)
p:<nome programma>	(Nome del programma di exit nella forma libreria/programma. Obbligatorio.)
s:<ampiezza campo>	(Ampiezza in caratteri del singolo campo del menù. default=32. Obbligatorio.)
m:<numero righe>	(Se inserito presenta il combo box aperto per il numero di righe specificate. Nel caso contrario il combo box è chiuso sull'opzione selezionata e può essere aperto con l'apposito tasto. Facoltativo.)

Il programma verrà fatto girare nel job applicativo (con il pgm principale in pausa dopo la EXFMT) se la sessione è BATCH mentre verrà fatto girare in un ambiente _temporaneo_ (solo con lista libreria uguale al job NETVT) se la sessione è INTERATTIVA.

Vedere **EXIT per riempimento dati keyword DDS SEV** in questo manuale per il dettaglio di funzionamento del programma di Exit.

Esempio:
se un campo ha dei valori definiti e i valori sono specificati in una tabella esterna è possibile creare un programma di exit che le reperisce e collegarlo al campo stesso.

Indicare nel campo testo dello stesso:
NA*SEV v:[TIPOCLIFOR](1) p:*LIBL/DECTIPOCF s:15

*SFC

Questa keyword a livello di campo permette di trasferire i possibili valori di un campo di scelta di un subfile dal campo stesso alla testata del subfile. Il campo stesso viene trasformato in un check box e diventa possibile la scelta multipla. Sopra la testata del subfile viene creato un pulsante per ogni scelta inserita.
Si possono inserire le descrizioni dei pulsanti sia fornendo la costante relativa oppure facendo riferimento ad un file messaggi residente su iSeries (per applicazioni multilingue). In alternativa si può utilizzare la keyword *SFP (vedere di seguito) che posizionata sui campi che esprimono le descrizioni le collega ai pulsanti del subfile.
Si può usare in coppia con la keyword CLS e in questo caso la classe specificata verrà usata per la generazione dei pulsanti. Il check box della riga subfile userà invece la classe _nasfc_class_ per valori vuoti oppure la classe _nasfc_value_class_.
La keyword può essere usata anche al di fuori di un subfile ed in questo caso il campo su cui viene posta viene trasformato in una sequenza di pulsanti.

Il numero massimo di pulsanti gestibili è 40.
Come parametri prevede:

f:<file messaggi>	(nome del file messaggi su iSeries che contiene le descrizioni dei valori. Formato: *LIBL/<nome>, <libr>/<nome>, <nome>. Facoltativo.)
<valore>=<descrizione>	(possono essere inseriti n valori separati da spazio. I valori devono avere la stessa lunghezza del campo. Se è stato indicato il parametro f: nella descrizione si indica l'id del messaggio; altrimenti nella descrizione si inserisce la costante da associare al valore. Previsti max 40 elementi.)
k:<tasti funzione>...	(può essere inserito il tasto da premere quando viene selezionato un pulsante. I valori ammessi sono Invio, F1 – F24, Rollup, Rolldown, Help. Se viene omesso il parametro viene simulato un Invio. Per indicare di non simulare alcun tasto indicare none.)
n:<n.righe>	(N. di righe su cui distribuire i tasti di comando. Deve essere espresso con due cifre. Se viene omesso spalma sulla base dell'ampiezza s:)
s:<ampiezza campo>	(Ampiezza in caratteri dei tasti di comando. Facoltativo – se non immesso ogni tasto assume l'ampiezza del testo contenuto.)

Esempio:
se un campo scelta di un subfile prevede le funzioni X=Varia, A=Annulla, R=Ripristina verrebbe creato automaticamente un menù di selezione con i valori indicati al posto del campo stesso.
Si può scegliere un modo di rappresentazione diverso facendo diventare le scelte un semplice check box e trasferendo i pulsanti di selezione sopra il subfile, rendendo la funzionalità più simile a quella comunemente usata su Internet. Indicare nel campo testo del campo scelta del subfile:
`NA*SFC X=Varia A=Annulla R=Ripristina`
La stessa versione riferita a file messaggi (con traduzione quindi) va indicata così:
`NA*SFC f:*LIBL/MSGAPPL X=USR0001 A=USR0002 R=USR0003`

---

*SFH

Questa keyword a livello di campo subfile permette di abilitare un campo nascosto per il subfile esteso oppure per dare la descrizione di testata o la modalità di editazione numerica ad un campo già visibile. La keyword ha senso solo all'interno di un subfile esteso (vedere keyword *SFX). Questa keyword funziona solo in modalità batch ed è ignorata in modalità interattiva.
Come parametri prevede:

d:<descrizione>	(diventerà la testata della riga del subfile)
f:<file messaggi>	(diventerà la testata della riga del subfile)
e:<mascheda editazione>	(nella forma EdtWrd **solo** per l'editazione dei campi numerici)
w:<larghezza>	(larghezza in pixel riferito a 800x600 della colonna. Facoltativo).
a:<descrizione aggiunta>	(descrizione usata nella maschera di editazione e spostamento delle colonne).

Esempio:
per abilitare un campo nascosto alla rappresentazione nel subfile esteso indicare nel campo testo dello stesso:
`NA*SFH d:"Quantità" e:"   .   ,  "`
Lo stesso con riferimento a file messaggi:
`NA*SFH f:*LIBL/XDMSGF d:MSG0024 e:"   .   ,  "`

*SFP

Questa keyword a livello di campo permette far diventare il contenuto del campo un pulsante per la keyword *SFC. Il contenuto del campo (costante o di output) deve essere nel formato x=yyy dove x è l'opzione e yyy è la sua descrizione. X dovrà corrispondere ad un valore del campo scelta e yyy sarà inserito fra i pulsanti azione disponibili. Se non espresso il contrario il campo stesso viene rimosso dal video. Sopra la testata del subfile viene creato un pulsante per la scelta espressa. Se il campo è condizionato, non è visualizzato o contiene spazi a runtime il pulsante non viene generato. Il massimo di pulsanti generabili per un subfile è 40. Sono eleggibili anche i campi nascosti (solo in modalità batch).
La keyword può essere usata anche al di fuori di un subfile con le stesse regole della keyword *SFC.

Riferimenti in configurazione per le *SFP e *SFR

CONFIG.XML

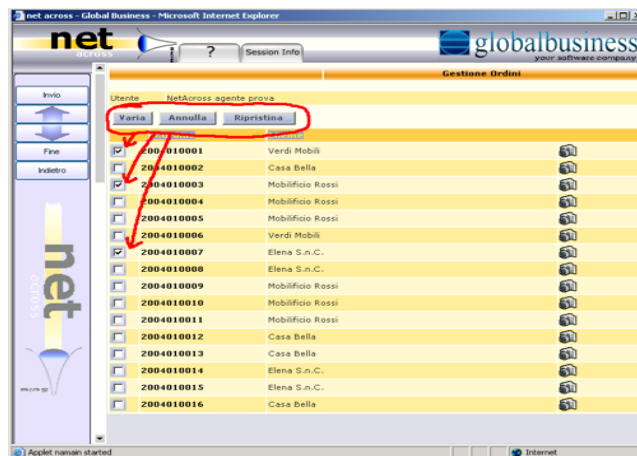
`<SfpSepDft>=</SfpSepDft>:` `Modifica il default del carattere separatore per la keyword, il default se non espresso è "="`

Questa keyword si differenzia dalla keyword *SFR perché prevede la ricerca di un unica scelta dentro il campo. In alcune applicazioni vengono utilizzati una serie di campi di output lungo tutta la riga e questi vengono riempiti a tempo di esecuzione con le scelte attive in funzione delle circostanze specifiche.

Come parametri prevede:

r:<nome campo>	(nome del campo su cui è posizionata la

d:N keyword *SFC. E' necessario se sono presenti nella pagina più keyword *SFC:) (se espressa evita la cancellazione del campo)

Esempio:
se una serie di campi che esprimono la descrizione delle scelte di un subfile prevedono le funzioni X=Varia, A=Annulla, R=Ripristina è possibile trasformare queste scelte in pulsanti tramite questa keyword.
Indicare nel campo testo di ogni singolo campo:
NA*SFP
Se nella pagina sono presenti più keyword SFC indicare:
NA*SFP r:V1SCEL

***SFR**

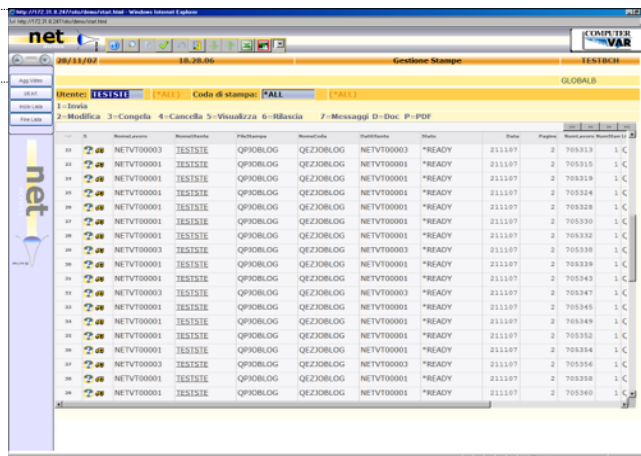
Le scelte dei subfile per la keyword *SFC possono essere assegnate facilmente con la keyword *SFP. Questo viene effettuato facilmente se le scelte del subfile sono rappresentate una per ogni campo. Se invece viene utilizzato un unico campo che contiene più scelte non si è in grado di riempire correttamente i pulsanti del subfile.
Questa keyword a livello di campo permette di indicare al generatore di Html che il campo di output rappresenta più descrizioni di scelta subfile e che il contenuto può essere utilizzato per riempire le descrizioni dei pulsanti relativi. Il campo una volta portato nei pulsanti viene eliminato dall'Html.
Questa keyword si differenzia dalla keyword *SFP perché prevede la ricerca di più tasti di comando dentro un unico campo. In alcune applicazioni viene utilizzato un unico campo di output lungo tutta la riga e questo viene riempito a tempo di esecuzione con i tasti di comando attivi in funzione delle circostanze specifiche.

Come parametri prevede:
r:<nome campo> (nome del campo su cui è posizionata la keyword *SFC. E' necessario se sono presenti nella pagina più keyword *SFC:)

Esempio:
se un campo di output conterrà a runtime la stringa 4=Annulla e 16=Ripristina indicare nel campo testo dello stesso:
NA*SFR
Se nella pagina sono presenti più keyword SFC indicare:
NA*SFR r:V1SCEL

*SFX

Questa keyword a livello di record subfile permette di abilitare la gestione del subfile esteso. In questa modalità le righe del subfile vengono gestite localmente sul client e quindi sono disponibili una serie di funzionalità aggiuntive rispetto ai subfile gestiti nel modo tradizionale.
Questa keyword funziona solo in modalità batch ed è ignorata in modalità interattiva.
Le differenze funzionali sono:
Dal punto di vista verticale è possibile scorrere le righe localmente nel browser una volta scaricate senza bisogno di richiedere le pagine nuovamente all'applicazione. Questo anche se il programma è scritto per riempire le pagine una alla volta. Una volta richiesta una pagina (tramite lo scorrimento o tramite i pulsanti veloci) le precedenti non vengono perse ma vengono accumulate a livello di browser. Tramite i pulsanti di scorrimento, i pulsanti veloci o l'ascensore a lato è possibile scorrere tutte le righe localmente ed in modo continuo.
Dal punto di vista orizzontale le righe si sviluppano verso destra senza il limite dell'ampiezza della pagina ma eventualmente generando l'ascensore orizzontale. Le eventuali righe di drop vengono aggiunte sulla destra trasformandosi in una riga unica. E' possibile anche portare nella riga visibile anche eventuali campi nascosti presenti nella riga del subfile (tramite la keyword *SFH, spiegata più avanti). Ogni utente ha poi la possibilità tramite un apposito pulsante di personalizzare quali campi fra i possibili devono essere resi visibili ed in quale ordine, in particolare per portare nel campo immediatamente visibile i campi più interessanti.
Questa keyword attualmente supporta al suo interno le keyword:
***LNK**, ***IMB**, ***IML**, ***IMG**, ***EVT**, ***SFC** e ***CLS**. Come widget supporta i **check box**. *Eventuali JavaScript personalizzati potrebbero funzionare in modo diverso rispetto ai subfile tradizionali.*
A livello di PTF verranno via via aggiunte nuove funzionalità sia a livello di keyword sia a livello di widget supportati.
Come parametri prevede:

h:<altezza delle righe>	(altezza in pixel).
n:<num.righe visibili>	(numero di righe visualizzate ,ignorato se < 3, il valore non può essere maggiore del buffer)
s:<scroller Y/N>	(indica se deve essere presente lo scroller laterale, per step inferiori alla pagina)
b:<righe di buffer>	(numero di righe che vengono utilizzate come buffer in download. Come default vale SFLPAG*2)

Esempio:
per abilitare un subfile al modo esteso indicare nel campo testo del record subfile:
NA*SFX

Suggerimento

Nel caso si utilizzi il subfile esteso con la barra di scorrimento, si potrà verificare il caso in cui il posizionamento a riga sul browser non sarà allineato con la paginazione (modalità 5250) per fare in modo che il programma RPG abbia l'informazione di ritorno su quale sia la prima riga effettiva visualizzata nel browser sono necessarie alcune operazioni.

Nel display file definire il subfile record number sul CONTROL del SUBFILE in questo modo:

```
A            SFLPO1          4S 0H      SFLRCDNBR(*TOP)
```

Nel programma RPG, la definizione del video deve contenere la INFDS:

```
FTESTPOSV  CF   E              WORKSTN USROPN
F                                      SFile(SFL:NRec1)
F                                      INFDS(ds_video)
```

e quindi:

```
d ds_video        ds
d  cursor               370    371b 0
d  sflrrn               376    377I 0
d  minrrn               378    379I 0
d  numrcd               380    381b 0
```

Impostando SFLPO1 si posizionerà il subfile a quella riga, mentre come valore di ritorno si avrà in MINRRN il primo record della videata.

Per abilitare la visualizzazione del numero di riga è necessario modificare il commands_defaults.js relativo alla start.html di avvio sessione inserendo:

```
var SFX_NRIG_COLUMN = 'RIGHT';  /* oppure 'LEFT' */
```

***SPL**

Questa keyword a livello di record permette di suddividere un record in più tabelle HTML. Tale keyword è utile quando il record presenta dati discontinui nel suo svolgimento, ad esempio una parte con dati normali ed un'altra con dati esposti in forma tabellare. Il generatore HTML cercherebbe di incolonnare i campi con un'unica logica proponendo un'interpretazione poco leggibile. Con questa keyword è possibile indicare a quale riga DSPF iniziare una nuova logica di incolonnamento campi, aprendo quindi una nuova tabella a livello di HTML. Il valore si riferisce alla riga (compresa) dalla quale effettuare la separazione.

Come parametro devono essere inseriti i valori di riga alla quale iniziare una nuova tabella:

<numero riga>	(obbligatorio e ripetibile n volte separato da spazio. Ammessi numeri riga da 2 a 27)

Esempio: per suddividere la logica di incolonnamento di un formato record con una prima parte compresa fra riga 1 e riga 10 ed una seconda compresa fra riga 11 e riga 24 indicare nel campo testo del record:
NA*SPL 11

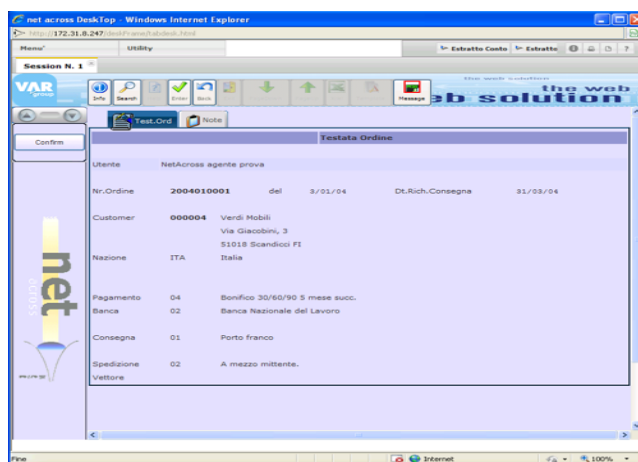
---

*TAB

Questa keyword a livello di record permette di raggruppare più pagine 5250 logicamente collegate in un unica pagina. La navigazione fra le pagine verrà effettuata con la logica delle linguette. **Disponibile solo in ambiente batch**.
Come parametri prevede:

[<*nome_campo*>](*ON/*OFF)	(Da ripetere n volte quanti sono i TAB. Specificare in <campo> la descrizione del tab. *ON per indicare il TAB corrente. Obbligatorio.)
o:<*campo_tab_out*>	(Campo hidden che riceverà il nome del TAB selezionato. Obbligatorio.)

La sintassi prevede **la ripetizione della sintassi NA*TAB per tutti i formati che compongono il gruppo** (settando *OFF e *ON il parametro che stabilisce quale TAB è attivo nel momento in cui è visibile il formato record su cui è applicata la regola) e di **un campo hidden per ogni formato** (lungo a sufficienza per contenere il testo descrittivo e il path dell'immagine) e **un campo hidden per il ritorno** della scelta effetuata dall'utente, lungo 10A in quanto dopo l'exfmt conterrà il nome campo relativo al tab selezionato.

Esempio:
se due record (testata ordine e note) sono raggruppabili logicamente in TAB indicare nel record della testata ordine:

```
A*NA*TAB [TAB1](*ON) [TAB2](*OFF) O:[TABO1]
```

e definire come campi hidden nel formato video TAB1/TAB2 e TABO1.
Indicare nel record delle note:

```
A*NA*TAB [TAB1](*OFF) [TAB2](*ON) O:[TABO1]
```

e definire come campi hidden nel formato video TAB1/TAB2 e TABO1.
A tempo di esecuzione riempire il campo TAB1 con l'immagine e la descrizione del TAB Testata Ordine (separati dal carattere l) ed il campo TAB2 con l'immagine e la descrizione del TAB Note.

E’ possibile specificare un’immagine e un testo da visualizzare nel tab, questi devono essere impostati valorizzando in questo modo le variabili associate ai tab:

```
Eval       TAB1 = 'file:/images/tord.gif|Test.Ord'
Eval       TAB2 = 'file:/images/notes.png|Note     '
```

Premendo il TAB “Test. Ordini” o “Note” verrà simulato il tasto “INVIO” ed il campo TABO1 conterrà il nome del formato che è stato richiesto dalla funzione TAB. Il programma gestionale dovrà provvedere ad emettere tale formato.

Per modificare la grafica del menu TAB è necessario personalizzare il file menuTab.css presente nella cartella /SCRIPTS/XDOM/RC. Inserendo il file nella cartella /STYLE/APPLICAZIONE per gli utenti registrati all’APPLICAZIONE nella Validation List verrà utilizzato questo file personalizzato.

Specifiche di compatibilità:
1) E' possibile abilitare un TAB per UN SOLO FORMATO se presente un gruppo di EXFMT con l'eccezione dei formati CONTROL+SUBFILE che vanno insieme.
2) Le coppie CTL+SFL si aspettano che il CONTROL sia superiore nell'ordine delle righe dall'alto verso il basso e abbia almeno un campo visibile.
3) E' possibile visualizzare un formato come testata unica per tutti i formati se questa è ripetuta in WRITE per ogni EXFMT dei formati con le keyword NA*TAB e i TAB saranno visualizzati sotto questo formato.

---

*UPL

Questa keyword a livello di campo permette di trasformare il campo in un pulsante che consente il caricamento di un file dal PC utilizzando la funzione sfoglia standard. Il campo può essere un campo fittizio inserito al solo scopo di posizionare il pulsante. Il valore della keyword sta nel fatto che il percorso dove verrà posizionato il file PC nel sistema viene determinato dalla keyword stessa e può derivare dai campi presenti a video. In questo modo un'eventuale reperimento di immagini da scanner o fotocamera potrà essere guidato e la disposizione delle stesse in archiviazione potrà seguire delle regole prestabilite. L'ampiezza massima del singolo archivio da trasferire può essere limitata a livello di server HTTP.
Opzionalmente si potrà dare modo all'utente di confermare o variare la posizione prestabilita nella finestra di acquisizione, in qual caso il programma si occuperà di recuperare dal campo indicato il valore di ritorno e memorizzarlo secondo logiche applicative.
Come parametri prevede:

v:[<campo>](<lunghezza>)	(obbligatorio e ripetibile n volte)
d:<campo di riferimento>	(opzionale. Indica che la path di destinazione è modificabile ed in questo campo verrà inserita la path modificata. Il campo deve essere di input/output.)
c:<conferma sostituzione>	(opzionale. Chiede eventualmente conferma di sostituzione se viene trovato

l'oggetto destinazione. Y/N. Default=N)

Esempio:

se dobbiamo caricare delle immagini articolo da una macchina fotografica occorre creare un campo fittizio che poi diventerà il pulsante Upload. A video dovranno essere presenti il o i campi che identificano il codice articolo dell'immagine da prelevare. Nel parametro v: non deve essere inserita l'estensione del file poiché questa verrà reperita dal file in ingresso.
Indicare nel campo testo del campo fittizio:

```
NA*UPL v:”/immagini/‶[CODART](6)[COLORE](5)
```

Se si vuole dare la possibilità di modificare la destinazione dell'immagine:

```
NA*UPL v:”/immagini/‶[CODART](6)[COLORE](5) d:PATHIM
```

Deve essere configurato l'Alias corrispondente per ricevere la funzione (vedere Gestione Alias/Directory in questo manuale).

*XLS

Questa keyword a livello di record subfile permette di abilitare o disabilitare il pulsante xls che permette di esportare il contenuto dello stesso in un foglio di calcolo.A livello di esecuzione se viene premuto il tasto xls associato al subfile lo stesso viene riempito completamente ed i dati vengono forniti in formato foglio di calcolo. _Questa keyword funziona solo in modalità batch ed è ignorata in modalità interattiva._
Come parametri prevede:

<abilitazione>	(valori Y o N. Facoltativo. Default è Y.)
v:<nome campo testata>	(Nome del campo del formato video esterno al subfile (es. testata) da passare al foglio elettronico. Si possono inserire più campi separati da virgola. Facoltativo.

Esempio:

se abbiamo il valore di default impostato nel non abilitare il pulsante xls per tutti i subfile e vogliamo abilitare questo subfile specifico indicare nel campo testo del record subfile:

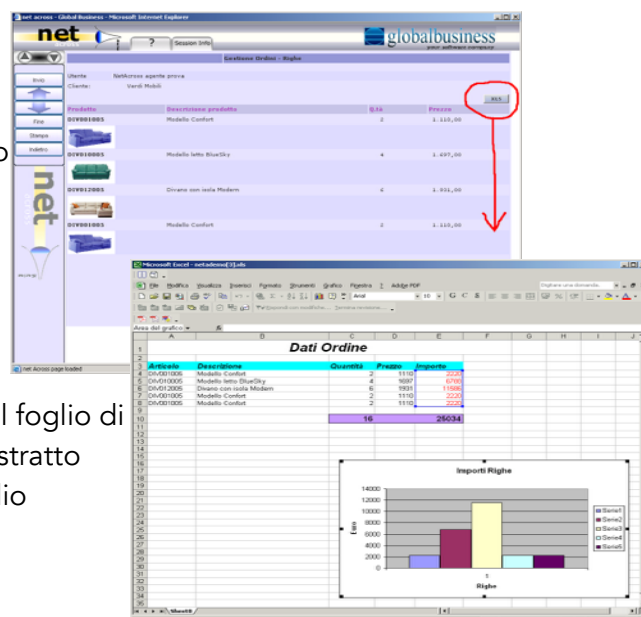
```
NA*XLS
```

oppure

```
NA*XLS Y
```

Se vogliamo abilitare il pulsante xls e vogliamo passare al foglio di calcolo i campi di testata nome cliente (RASCLI) e data estratto conto (DATAEC) per visualizzarli nell'intestazione del foglio elettronico indicare nel campo testo del record subfile:

```
NA*XLS Y v:RASCLI,DATAEC
```

Deve essere configurato l'Alias corrispondente per scrivere ed inviare i file di lavoro vedere Gestione Alias/Directory in questo manuale.
E’ possibile produrre dei template specifici per il singolo subfile inserendo gli stessi nella struttura

del file system: /netacross/XLS
La sottocartella successiva corrisponde alla APPLICAZIONE così come definita sul profilo utente di validation list .
La sottocartella successiva è il nome della LIBRERIA che contiene l'oggetto DSPF di riferimento
Il nome del file template deve essere NOMEFILEVIDEO.NOMEFORMSFL.xls
Alla pressione del tasto [xls] sulla videata l'automatismo genererà un xls partendo da quello configurato nella posizione indicata.

NEW! dicembre 2014

E' possibile indicare il nome del foglio di excel in estrazione utilizzando un campo H (hidden) a cui è assegnata la classe speciale .**XLSFILENAME**. Il valore che il campo assume a programma verrà utilizzato come nome de l file xls (senza estensione)

Riferimenti in Config.xml

<ToXLS>[Y/N]<ToXLS>: default di abilitazione del pulsante di estrazione per ogni subfile, viene letto in fase di conversione.
<XlsAlias>*nome_alias*</XlsAlias>: alias di riferimento per i fogli di excel generati tramite estrazione dai subfile, deve esistere nella gestione Alias (MENU CONF Opz.21)
<GestAuth>[S/N]</GestAuth>: abilitazione delle autorizzazioni alle estrazione di excel e al riordino delle colonne

---

*WIN

Questa keyword a livello di record permette di forzare la creazione di una window anche se il record DSPF non lo è. In particolare questa keyword è utile quando i programmi AS400 generano delle finte windows (OVRATR, OVRDTA, ecc.) ma a livello di HTML si vuole avere una resa da window effettiva. La keyword deve essere inserita per tutti i formati video che compongono la finestra.
Come parametro devono essere inseriti i valori di riga iniziale, colonna iniziale, ampiezza righe, ampiezza colonne (esattamente con ampiezze e spazi come indicato):

(rr ccc rr ccc) (obbligatorio. Ammesse righe e colonne entro l'ampiezza dello schermo 5250)

Esempio:

per forzare un record normale in una window che inizia alla riga 3 colonna 5 e ampia 10 righe e 40 colonne indicare nel campo testo del record:

```
NA*WIN (03 005 10 040)
```

Funzioni JavaScript di NetA

L'applicazione utilizza pagine Html per veicolare le informazioni richieste all'iSeries. Tali pagine possono essere rese attive assegnando ad eventi segmenti di codice eseguibile.
Un linguaggio molto diffuso per questo scopo è il JavaScript. Tale linguaggio ha una propria sintassi ed ha a disposizione i dati, gli eventi e i metodi della pagina Html su cui viene attivato.
Esistono in commercio e su Internet moltissimi manuali ed esempi di programmazione in questo linguaggio.
Insieme a NetAcross viene fornita una libreria di funzioni JavaScript per facilitare l'utilizzo delle operazioni connesse all'ambiente specifico. Ogni utente può comunque integrare o modificare tali funzioni per adattarle alle proprie esigenze.

Ecco alcune regole valide per le funzioni che sono fornite:

- Gli esempi seguenti riguardano il testo JavaScript dell'Html da inserire nella pagina, in quanto a tempo di esecuzione ad esempio gli eventuali *value saranno, dove è previsto, sostituiti dal valore dinamicamente calcolato.

- L'uso del doppio o del singolo apice è condizionato da dove è inserito il testo del JavaScript. Nel caso dell'uso delle keyword TEXT NA*, non è conveniente inserire l'apice singolo a causa della sintassi DDS, per cui è consigliabile il carattere ("). Sarà il generatore a convertire i caratteri durante la generazione della pagina Html.

- In Javascript sono ammessi entrambi i caratteri a delimitare le stringhe, naturalmente ad un apice/doppioapice aperto deve corrispondere un coerente apice/doppioapice per chiudere il testo.

Esempio:

```
javascript:onclick="NetAcross.setValue('CODCLI',*value);"
```

Nel frame “main” all'interno della pagina NetAcross sono disponibili delle variabili che possono essere utili in diversi casi. Ad esempio si può cambiare il marchio aziendale in funzione del sistema informativo scelto o evidenziare sessioni su archivi effettivi o di prova.

Le variabili sono:

NetA.sisinf	Sistema informativo (solo modalità batch e modulo base ACG)
NetA.menuini	Menù iniziale utente (solo modalità batch)
NetA.netvt	Nome della sessione iSeries (nome terminale assegnato)
NetA.naid	ID della sessione iSeries (numero progressivo)
NetA.nausr	Nome dell'utente collegato
NetA.usertype	Tipo utente (solo per utenti di Validation List)
NetA.lang	Linguaggio assegnato al profilo utente
NetA.iseries	Nome del sistema iSeries. E' il nome macchina e non il nome host

NetA.nlib Nome Libreria del display attivo
NetA.nfil Nome File del display attivo
NetA.nfmt Nome Formato Record del display attualmente in "Execute Format"
NetA.vldl Nome della validation list che ha validato l'utente

NetA.F("*nome_tasto_funzione*");

Questa funzione permette di simulare la pressione di un tasto funzionale. Si può quindi automatizzare l'uscita da una pagina in funzione di eventi particolari. Ad esempio nel caso di un subfile a scelta singola si può aggiungere alla selezione della voce la generazione di un invio automatico agganciandolo all'evento onclick oppure onchange.
Come parametro prevede: Invio, RollUp, RollDown, Help, F1->F24.

Esempio:

per eseguire il tasto Invio:
```
F("Invio");
```
per eseguire il tasto F15:
```
F("F15");
```

NetA.setValue("*nome_campo*","*valore*")

Questa funzione permette di impostare un campo dati della pagina Html con un valore costante o con il risultato di una operazione *Value (vedere keyword DDS in questo manuale).
E' utile quando si vuole forzare un campo ad un valore specifico oppure si vogliono unire valori provenienti da più campi al verificarsi di un evento.
Come parametri prevede:

nome_campo	Nome del campo a cui assegnare il valore (fra apici) Se campo di subfile e all'interno di una keyword DDS il nome varia in funzione della riga, e quindi deve essere indicato con la forma speciale ***name:***nome_campo******
valore	Valore da assegnare (fra apici) oppure il valore speciale *value per prendere il valore calcolato a runtime dalle keyword DDS.

Esempio:
```
NetA.setValue("CODCLI","000001")
```
Assegna il valore 000001 al campo CODCLI.

Esempio:
```
NetA.setValue("CODCLI",*value)
```
Assegna il valore dinamico calcolato a run-time dalle keyword DDS al campo CODCLI.

Esempio:
```
NetA.setValue(*name:CODCLI*,*value)
```
Assegna il valore dinamico calcolato a run-time dalle keyword DDS al campo CODCLI interno alla riga specifica del subfile.

NetA.putCursor("*nome_campo*")

Questa keyword forza la posizione del cursore in un determinato campo oppure, solo per la modalità interattiva, in una posizione riga/colonna con la sintassi RxxCyyy.
E' utile quando si eseguono funzioni sensibili alla posizione del cursore 5250, ma nella pagina 5250 non esistono campi di immissione corrispondenti. Ad esempio se esiste una selezione di una riga di un subfile sulla base della posizione cursore e non esiste un campo di input si può aggiungere un pulsante per la riga che posiziona il cursore sulla stessa e preme Invio.
Come parametri prevede:

nome_campo	Il nome del campo che riceverà il cursore nella prima posizione (fra apici). In modalità interattiva il campo si può esprimere nella forma RxxCyyy per indicare una riga ed una colonna.

Esempio:

```
NetA.putCursor("CODCLI")
```

Il cursore viene posizionato sul primo carattere del campo CODCLI.

NetA.get({*path_composito*})

Questa keyword permette di ottenere un file dal server iSeries. Il file viene fornito attraverso il meccanismo degli Alias (vedere Gestione Alias/Directory in questo manuale) e quindi può risiedere anche su altri server della rete. Non necessita l'apertura delle cartelle nel server Http.
E' compito del browser decidere cosa fare del file scaricato. Se c'è un plug-in installato corrispondente questo viene aperto automaticamente. Altrimenti viene richiesto se aprire o salvare il file.
Come parametri prevede:

path_composito	Il percorso + nome file dove risiede il file da scaricare.

Esempio:

```
NetA.get("/PDF/AGENTI/"+*value+".pdf")
```

Permette di scaricare il file che si trova nella alias PDF, sottodirectory AGENTI con il nome dinamicamente calcolato da una keyword DDS e con estensione pdf (formato MIME application/x-pdf).

NetA.getImage({*path_composito*})

Questa keyword permette di ottenere un file contenente immagini dal server iSeries visualizzando il risultato in una finestra separata e dimensionata all'immagine stessa. La funzione ha le stesse regole della funzione NetA.get. Il file viene fornito attraverso il meccanismo degli Alias (vedere Gestione Alias/Directory in questo manuale) e quindi può risiedere anche su altri server della rete. Non necessita l'apertura delle cartelle nel server Http.
Come parametri prevede:

path_composito	Il percorso + nome file dove risiede il file da scaricare.

Esempio:

```
NetA.getImage("/IMMAGINI/ARTICOLI/"+*value+".jpg")
```

Permette di visualizzare l'immagine che si trova nella alias IMMAGINI, sottodirectory ARTICOLI con il nome dinamicamente calcolato da una keyword DDS e con estensione jpg (formato MIME image/jpg).

NetA.getFile({*path_composito*})

Questa keyword permette di ottenere un file dal server iSeries con download automatico. ~~visualizzando il risultato in una finestra separata con la richiesta di apertura o salvataggio.~~ La funzione ha le stesse regole della funzione NetA.get. Il file viene fornito attraverso il meccanismo degli Alias (vedere Gestione Alias/Directory in questo manuale) e quindi può risiedere anche su altri server della rete. Non necessita l'apertura delle cartelle nel server Http.
Come parametri prevede:

path_composito	Il percorso + nome file dove risiede il file da scaricare.

Esempio:
```
NetA.getFile("/NADOCS/XLS/"+*value+".xls")
```
Permette di ~~aprire o~~ salvare il foglio di calcolo che si trova nella alias NADOCS, sottodirectory XLS con il nome dinamicamente calcolato da una keyword DDS e con estensione xls (formato MIME application/vnd.ms-excel).

NetA.downloadPdf({*path_composito*},"*tasto continuazione*")

Questa keyword permette di ottenere un file di tipo PDF dal server iSeries visualizzando il risultato in una finestra separata a cura del plug-in relativo. La funzione richiama la funzione NetA.get aggiungendo delle funzioni specifiche legate al plug-in di renderizzazione PDF. In particolare il meccanismo serve a dare il focus alla finestra del PDF e a chiudere automaticamente la finestra di attesa dell'esecuzione quando tale finestra viene chiusa dall'utente. Vedere Lancio automatico di documenti PDF nel manuale Installazione e Configurazione.
E' utile quando si vuole visualizzare un PDF che si è creato al momento (ad esempio al termine dell'inserimento di un ordine) e occorre attivare dei meccanismi di attesa della creazione della stampa e della sua trasformazione in PDF. Se si vogliono visualizzare dei PDF già memorizzati è appropriata la funzione NetA.get.
Il file viene fornito attraverso il meccanismo degli Alias (vedere Gestione Alias/Directory in questo manuale) e quindi può risiedere anche su altri server della rete. Non necessita l'apertura delle cartelle nel server Http.
Come parametri prevede:

path_composito	Il percorso + nome file dove risiede il file da scaricare.
tasto continuazione	Viene premuto questo tasto quando si chiude il PDF. Può valere "Invio" o "Fxx". Se non viene inserito vale "Invio".

!Attenzione!: la pressione del tasto in chiusura documento potrebbe non essere supportata dai visualizzatori PDF, quindi la funzione potrebbe operare non in modo previsto.

Esempio:
```
NetA.downloadPdf("/PDF/AGENTI/"+*value+".pdf")
```
Se questa funzione è agganciata ad una keyword *EVT per esempio, permette di visualizzare l'immagine che si trova nella alias PDF, sottodirectory AGENTI con il nome dinamicamente

calcolato da una keyword DDS e con estensione pdf (formato MIME application/x-pdf).

NetA.setOnTool(toolbar_button_id);

Permette di attivare un pusalnte inserito nella toolbar, il parametro è l'id del bottone che si vuole abilitare. Nel caso di funzioni personalizzate gestite di volta in volta dalle pagine o autorizzate in base a particolari condizioni è possibile gestire la toolbar, che naturalmente deve essere stata disegnata correttamente in precedenza, la toolbar di solito viene posizionata nel frame top personalizzato.

NetA.setOffTool(toolbar_button_id);

Permette di disattivare un pusalnte inserito nella toolbar, il parametro è l'id del bottone che si vuole disabilitare. Vedi sopra.

NetA.sev_Ajax("*nome_campo*");

La funzione sev_Ajax permette di caricare i dati di una campo su cui è impostata la parola chiave *SEV , cioè un campo del video i cui valori obbligati sono espressi da un menù a discesa caricato tramite un exit program. Come indica il nome, i valori vengono caricati in maniera asincrona rispetto alla pagina, di solito per esempio, in conseguenza a scelte dell'utente è possibile caricare la lista delle opzioni in un secondo campo e così via.

Esempio:

Il campo V1CDCL viene caricato con il programma PGMSEV e quando l'utente esegue una scelta (evento onchange) vengono caricate 2 liste sui campi V1CDFOR e V1AGE.
Ogni campo ha la sue exit agganciata e riceve un parametro composto dai parametri precedenti, la sev_Ajax stimola il caricamento del campo indicato e ne popola i dati tramite il ritorno.

```
A*NA*SEV v:[V1PARM](1) p:*LIBL/PGMSEV1 s:20
A*  *EVT t:onchange j:§NetA.sev_Ajax('V1CDFOR');NetA.sev_Ajax('V1AGE')
A            V1CDCL         6A  B  6 23
 ...
A*NA*SEV v:[V1PARM1](1)[V1CDCL](6) p:*LIBL/PGMSEV2 s:20
A*  *EVT t:onchange j:§NetA.sev_Ajax('V1AGE')
A            V1CDFOR        6A  B 10 23
 ...
A*NA*SEV v:[V1PARM1](1)[V1CDCL](6)[V1CDFOR](6)p:*LIBL/PGMSEV3 s:20
A            V1AGE          4A  B 12 23
```

NetA.toTextArea("class_name" , window);

La funzione .toTextArea permette di creare un'area di testo libero, appunto una TEXTAREA html, come somma di tutti i campi che hanno impostata una particolare classe (tramite la keyword *CLS). In pratica si possono trasformare n campi di lunghezza uguale in un unico campo con caratteri a passo fisso.
I campi devono essere tutti di input e tutti della stessa lunghezza.

La funzione deve essere richiamata all'interna della funzione utente **function initpage() {...}** definita nel file javascript personalizzato per la pagina , inserito tramite la keyword *JSC.
Ed è possibile create più textarea raggruppando campi diversi identificandoli con classi diverse.
Esiste una classe speciale che attiva la funzione .toTextArea direttamente dal motore netacross nel momento in cui la pagina viene visualizzata, se si imposta *CLS v:CLATXAREA sui campi interessati , questi verranno trattati come una unica textarea in automatico senza dover richiamare la funzione.

NetA.googleMap("*indirizzo espresso in modo testuale***");**

La funzione permette di aprire una mappa indicando l'indirizzo in formato testuale come se l'utente facesse una ricerca client tramite google maps. La funzione non utilizza i servizi app di google, è il client che esegue la richiesta. Viene aperta una nuova finestra o un nuovo tab nel caso della visualizzazione in deskFrame.

NetA.viewCalendarFor("*nome_campo***" , "***formato_data***");**

La funzione permette di attivare la visualizzazione di un calendario sul campo indicato nel formato data specificato, in conseguenza per esempio di un evento gestito con la *EVT, clikkando su un'immagine o direttamente su un campo.

```
A*NA*EVT t:onclick j:$NetA.viewCalendarFor("LLDT1V","gg/mm/aa");
A            LLDT1V          6Y 0B   4   8EDTCDE(Y)
```

NetA.setRollUp("*nome_formato_record***");**

La funzione permette di simulare un tasto roll up per un formato record, l'utilizzo è previsto se non c'è il subfile esteso, per pilotare lo scorrimento tramite pressione di campi con *EVT

NetA.setRollDown("*nome_formato_record***");**

La funzione permette di simulare un tasto roll down per un formato record, l'utilizzo è previsto se non c'è il subfile esteso, per pilotare lo scorrimento tramite pressione di campi con *EVT

NetA.setTitle("*nome_campo***","***titolo***");**

La funzione permette di impostare su un campo i dati aggiuntivi visibili con una tooltip se il browser lo supporta

NetA.showImage("*path***");**

La funzione permette di visualizzare una finestra con l'immagine indicata nel path, NON utilizza gli alias (vedi getImage), quindi permette la visualizzazione di immagini esposte dall'http server direttamente.

showInfo();

Questa funzione permette di visualizzare il pannellino delle informazioni sul lavoro in esecuzione.

Dalla pagina 5250 è possibile estrarre una serie di informazioni se disponibili (titolo, nome del programma, ditta, ecc.) tramite la keyword DDS NA*INF. Queste informazioni vengono inserite nel pannellino a cui fa riferimento questa funzione.
Lo scopo del pannellino è isolare queste informazioni ma anche quella di poterle cancellare nella pagina Html per guadagnare spazio utile. Le informazioni anziché essere visualizzate costantemente potranno essere richiamate solo al bisogno.
Normalmente questa funzione è disponibile come pulsante nel frame di testata.
La funzione è resa disponibile dalla pagina principale in cui viene visualizzata la sessione.
La funzione non prevede parametri.
E' inserita automaticamente nella deskView.

---

initpage();

Funzione evento, che viene chiamata sulla pagina html ad ogni pagina nuova emessa dal server, in una fase iniziale della costruzione della stessa, cioè quando l'html base è stata creata ma prima degli automatismo. Deve essere definita nel javascript associato al formato video inserito con keyword *JSC e permette di eseguire codice prima che l'attività normale della pagina sia disponibile all'utente.

---

loadpage(); NEW! novembre 2014

Funzione evento, come la initpage() viene chiamata in una fase più avanzata della costruzione della pagina, dopo gli automatismi principali, e dopo la attivazione/disattivazione dei tasti funzionali

---

onPageLoad();

E' un evento speciale di netacross richiamato sulla pagina **top.pers** ad ogni emissione (EXMFT) di un formato video. Permette di eseguire codice personalizzato sulla top.pers della pagina html recuperando dati dalla pagina applicativa ed integrare altri dati in base alle necessità dell'ambiente.

---

onValueChange();

E' un evento speciale di netacross richiamato sulla pagina top.pers quando nel menù di architettura si cambia la scelta sistema informativo, cioè quando viene ricaricata la lista delle azioni disponibili per l'utente. Permette quindi di elaborare una personalizzazione in base al sistema informativo (per esempio cambiare il logo dell'azienda scelta). Vedi esempio sotto.

---

formValidation(doc); NEW! novembre 2014

E' una funzione richiamata in fase di submit del modulo nel momento in cui è stato premuto un tasto di aggiornamento del form e deve ritornare un boolean **true** o **false** , in base al quale ,se false, il modulo può essere bloccato nella fase di invio, l'eventuale messaggio di annullamento deve essere prodotto nella funzione stessa. Può essere definita nella libreria javascript generale o

nelle singole pagine con la *JSC

formConfirm(doc,fkey); NEW! novembre 2014

E' una funzione richiamata in fase di submit del modulo per ogni tasto premuto, anche non di aggiornamento e deve ritornare un boolean **true** o **false** . Se ritorna false il modulo non verrà inviato. La modalità di conferma è a carico dello sviluppatore, la funzione può essere definita nella libreria javascript generale oppure tramite la *JSC.

NetA.confirm(parms); NEW! novembre 2014

Permette di emettere una dialog di conferma e controllare la scelta dell'utente, eseguendo un tasto funzione
Si attiva per default se il campo di classe "selettore" contiene il valore "X" e ritorna nello stesso campo. Il campo di riferimento deve essere un campo B non di subfile.
I parametri sono:

parms.selector = classe di selezione campo, per recuperare il campo hidden di riferimento, [default se non impostato ".CONFIRM"]
parms.requestVal = valore di confronto per attivare la dialog , [default "X"]
parms.title = titolo della finestra
parms.message = messaggio informativo
parms.confirmVal = valore del campo ritornato per OK [default "K"]
parms.confirmLabel = descrittore del OK [default "OK"]
parms.confirmKey = tasto premuto per OK [default "Invio"]
parms.cancelVal = valore del campo ritornato per Cancel [default "N"]
parms.cancelLabel = descrittore del Cancel [default "Cancel"]
parms.cancelKey = tasto premuto per Cancel [default "Invio"]

NetA.alert(parms); NEW! novembre 2014

Permette di emettere una diamo di alert ed eseguire un tasto funzione.
Si attiva per default se il campo di classe "selettore" contiene il valore "X" e ritorna nello stesso campo. Il campo di riferimento deve essere un campo B non di subfile.
I parametri sono:

parms.selector = classe di selezione campo, per recuperare il campo hidden di riferimento, [default se non impostato ".ALERT"]
parms.requestVal = valore del campo per attivare la richiesta [default "X"]
parms.title = titolo della finestra
parms.message = messaggio informativo
parms.confirmVal = valore del campo ritornato a pressione bottone [default "K"]
parms.confirmLabel = descrittore del bottone
parms.confirmKey = tasto premuto a pressione del bottone [default "Invio"]

Esempio:

Nell'esempio si usa la funzione loadpage() per attivare l'alert quando la pagina è pronta, successivamente vengono recuperati i valori da presentare nella finestra prendendoli dai campi hidden con *CLS predisposti nella videata.
Infine viene richiamata la confirm (la alert è commentata)
Quando l'utente risponde alla richiesta verrà eseguito il default "Invio" e verrà valorizzato il campo di classe ".CONFIRM" con il valore "K" di default.

```
function loadpage(){
 var messaggio = $(".MESSAGE").val();
 var titolo    = $(".TITLE").val();
 var oklabel   = $(".OKLABEL").val();
 var okfunc    = $(".OKFUNC").val();

 NetA.confirm({message: messaggio, title: titolo, confirmLabel: oklabel});

 //NetA.alert({requestVal: "A", confirmKey: okfunc});

 };
```

Nel sorgente video i campi sono definiti così:

```
A*NA*CLS V:CONFIRM ALERT
A            ALERT          1A  B  4 10
A*NA*CLS V:MESSAGE
A            AMSG         100A  H
A*NA*CLS V:TITLE
A            ATIT          50A  H
A*NA*CLS V:OKLABEL
A            AKLB          50A  H
A*NA*CLS V:OKFUNC
A            AKFU          50A  H
```

Alcuni esempi di Javascript di uso comune:

Esecuzione di un'applicazione locale del PC (solo Windows)

Questo esempio apre l'applicazione Notepad in locale sul PC passando un parametro contenente il nome del file da aprire.
E' necessario abbassare i livello di autorità del browser per l'indirizzo da cui si scarica questo script, altrimenti la richiesta viene bloccata.
Questo script deve essere memorizzato in un file di tipo .JS e deve essere reso disponibile con la keyword *JSC al formato record interessato.

La funzione FuncTest(); che segue può essere richiamata per esempio con una keyword *EVT su un campo della pagina cliccando con il mouse.
E' stato immaginato un campo a video di nome FILEX che deve contenere il nome del file da aprire con il NotePad. Il campo deve contenere la keyword *INF per renderlo gestibile dal JavaScript.
Esempio:

```
function FuncTest()
  {
   var oShell = new ActiveXObject("Shell.Application");
   var commandtoRun = "notepad.exe";
   var commandParms = NetA.filex.value;
    oShell.ShellExecute(commandtoRun, commandParms, "", "open", "1");
 }
```

Richiesta obbligatoria utente/password al collegamento

Questo ci si collega al sito NetAcross l'utente e la password viene richiesto solo la prima volta, fino a quando non si chiudono tutte le sessioni del browser.
Aggiungendo il seguente JavaScript alla pagina di collegamento (start.html) è possibile ad ogni connessione anche senza la chiusura del browser di far emettere la richiesta di utente e password. Se era stata richiesta la memorizzazione della password questa viene comunque ripresentata.
Questa funzione funziona solo per Internet Explorer.

```
<script language="javascript" type="text/javascript">
try {
      document.execCommand("ClearAuthenticationCache", false);
   }catch(e){}
</script>
```

Modifica immagini/dati del Top del video

La parte Top della pagina NetAcross rimane fissa durante tutta la durata della sessione. E' possibile fare in modo che rispecchi alcuni stati per rendere evidente all'utente in quale posizione si trova.
Ad esempio è possibile far apparire un'immagine in funzione del sistema informativo selezionato.
NetAcross imposta una serie di valori di sessione, e l'elenco è nel capitolo precedente.
Nell'esempio che segue si recupera il valore del sistema informativo (NetA.sisinf) e si cambia il valore del campo Html di tipo immagine sinf_img facendolo puntare ad una immagine nella cartella <root>/images con lo stesso nome ed estensione jpg.
Questo JavaScript va inserito nel file Top.html. La funzione deve avere questo nome specifico.

```
function onValueChange(){
          id_sinf_img = document.getElementById("sinf_img");
          src = "images/" + NetA.sisinf + '.jpg';
          id_sinf_img.setAttribute("src",src);
}
```

Menù di Configurazione NetAcross

Per eseguire i menù di Netacross richiamarli col comando:

Call NETAP/NETAMENU <opzione>

dove opzione può valere:
- **CONF** = configurazione.
- **SVIL** = sviluppo.

Tale funzione provvede ad amministrare da solo la lista delle librerie.

Funzione CONF

Da questo menù è possibile manutenere gli utenti, visualizzare le connessioni attive e le log.

```
NETACONF                     Menù Configurazione                 Sist.  S066C924
                                                                 Utente ALE

  Anagrafiche
     1. Gestione Iscrizione Utenti
     2. Gestione Certificati (SSL)

  Interrogazioni
    11. Interrogazione Utenti Attivi
    12. Interrogazione Log

  Gestioni utility
    21. Gestione Alias/Directory       27. Gestione Servizi (*COM)
    22. Gestione Funzioni Estese
    23. Gestione Alias Librerie
    24. Creazione CSS da CSS base
    25. Gestione Subfile Esteso
    26. Gestione Autorizzazioni
    90. Scollega
===> _

F3=Fine   F4=Richiesta   F9=Duplicaz.   F12=Annull.
F13=Supporto informativo  F16=Menu principale del sistema
```

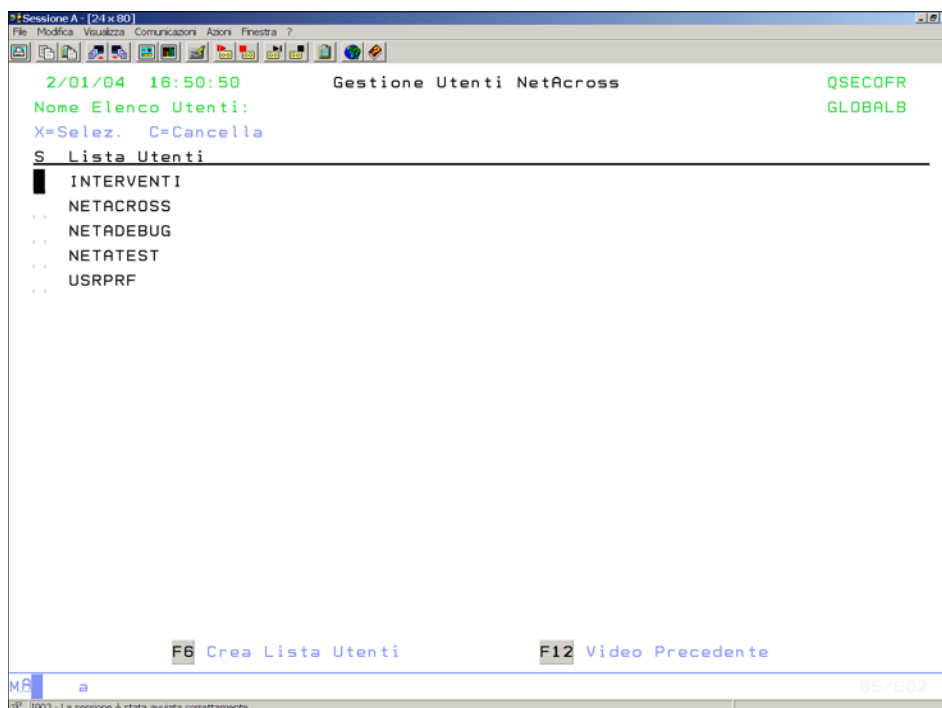
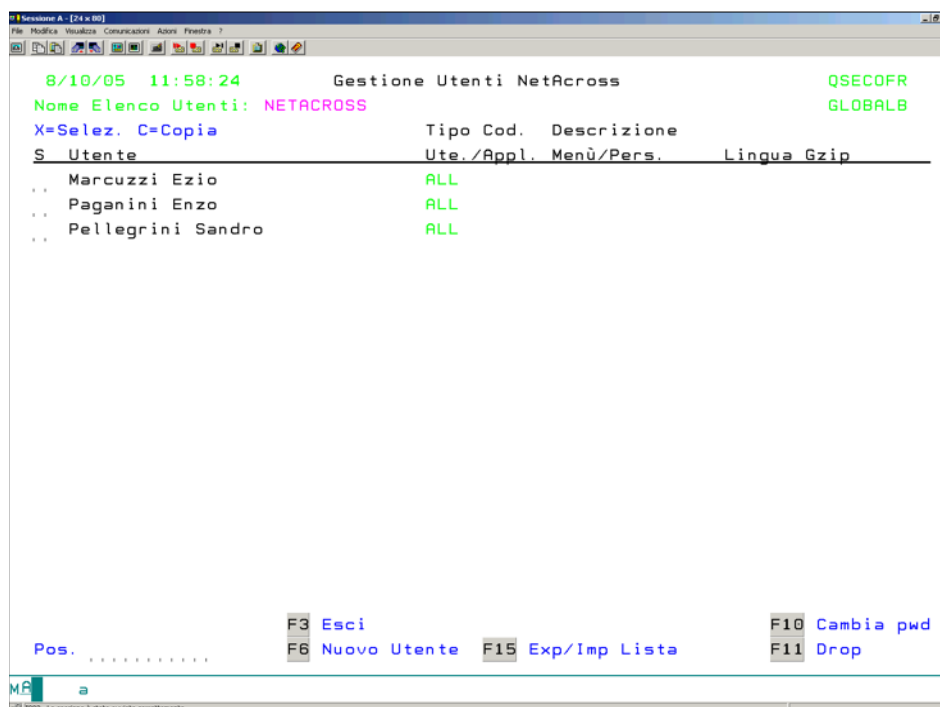
Gestione Iscrizione Utenti [1.]

Questa voce di menù permette di iscrivere gli utenti per l'utilizzo con NetAcross.
A livello di Http Server è possibile configurare l'istanza per validazione attraverso Validation List oppure tramite User Profile.

- La prima casistica si adotta quando si vuole consentire l'accesso ad utenti esterni all'azienda (Internet) ed è più pratico configurare degli utenti leggeri che non dei profili di sistema operativo.
- La seconda casistica si adotta quando l'accesso deve essere consentito ai normali utenti dell'azienda già registrati (Intranet).

Nel primo caso gli utenti sono registrati in particolari oggetti di tipo *VLDL, registrati nella libreria NETAF che memorizzano utente/password (criptata) e una serie di altri dati funzionali. Tali Validation List devono avere lo stesso nome dell'istanza Http.
Se si sceglie invece di collegarsi con utenti di sistema sarà comunque necessario includere gli utenti di sistema abilitati in una Validation List particolare chiamata **USRPRF** sempre gestibile da questo menù. Questo risulta necessario perchè nella Validation List vengono memorizzate informazioni relative all'utente collegato che altrimenti non sarebbero disponibili per gli utenti di sitema.

Ad ogni Validation List viene concessa l'autorizzazione in lettura da parte dell'Utente QTMHHTP1.
Alla Validation List speciale USRPRF viene concesso l'accesso lettura da parte di QTMHHTTP e alle altre l'accesso da parte dell'utente QPGMR.

Selezionando l'opzione 1 si accede alla gestione Validation List presenti e ne viene presentato l'elenco. Un tasto di comando (F6) ne permette la creazione.

Selezionando una Validation List è possibile accedere all'elenco degli utenti:

Eccetto la Validation List USRPRF è disponibile il tasto di comando F10 che consente di

modificare la password dell'utente di collegamento per tutti gli utenti iscritti nella lista.
Per tutte le Validation List è disponibile il tasto di comando F15 che permette di esportare o importare il contenuto della Validation List verso/da un file di database. Questo permette di effettuare delle modifiche massive o delle stampe utilizzando i normali strumenti di database. L'importazione prevede che gli utenti debbano essere già esistenti.

Per ogni utente i dati corrispondenti sono (eccetto la Validation List USRPRF che vedremo più avanti):

Al momento dell'accesso all'utente viene richiesto di autenticarsi. A tal fine viene richiesto in maniera codificata l'inserimento di un utente e di una password.
Utente Web (obbligatorio): è il nome utente richiesto al momento della validazione Web.
Password (obbligatorio): è la password richiesta al momento della validazione Web.
Tenere conto che entrambi i valori sono sensibili al maiuscolo/minuscolo.

L'utente che ha acceduto all'ambiente deve essere collegato ad un codice del gestionale per limitare l'accesso ai dati di propria pertinenza. Occorre quindi determinare il tipo ed il codice anagrafico. E' responsabilità dell'applicativo reperire tale dato attraverso interfacce NetAcross e limitare la disponibilità dei campi e/o tasti comando.
Il tipo ALL indica un utente che ha tutti gli accessi all'interno del menù assegnato, e quindi fa riferimento ad un uso di personale interno all'azienda.
Tipo (obbligatorio): indica la tipologia dell'accesso:

CLI = Cliente.
FOR = Fornitore.
AGE = Agente.

ALL = Tutti.
Codice (obbligatorio se != ALL): il codice nel gestionale del cliente/fornitore/agente.
Descrizione: un campo descrittivo dell'utente.

Per ogni utente collegato viene creata una sessione che esegue il lavoro iSeries. In questo paragrafo vengono specificati i dati per il collegamento che determina l'ambito di operatività (menù).
Si può indicare anche un nome da assegnare, cosa che facilita l'assegnazione di eventuali stampanti o funzioni automatiche.
Logon Utente/Password/Menù/Programma (obbligatorio ut/pwd): sono i dati per la connessione alla sessione. La password deve essere quella corrispondente all'user profile nel caso di accesso interattivo (vedere più avanti il parametro ENV); nel caso di accesso batch la password non è significativa e può essere inserito qualsiasi valore (eccetto password vuota).
Terminale: è il nome da usare per la sessione. Se non esiste sarà creato. Se non viene indicato verrà usato un nome nella forma NETVT<progr>.

I dati trasmessi dall'Http Server sono in formato HTML, un formato testo molto discorsivo. Le pagine Html acquistano spesso ampiezze rilevanti, compromettendo quindi la velocità ed i tempi di risposta. Un algoritmo di compressione molto efficiente permette di ridurre tale tempi in maniera drastica.
Compressione GZip: Y=abilitata N=Disabilitata A=Auto (da XML di configurazione). Per utilizzo Intranet locale è preferibile disattivare la compressione dei dati.

Gli iSeries prevedono diversi ambienti di funzionamento relativamente alle modalità cosiddette Interattive e Batch. NetAcross può funzionare in entrambe le modalità, ognuna con diverse considerazioni da tenere presenti. Nella modalità interattiva NetAcross interagisce con l'applicativo intercettando l'Input/Output del terminale e non sono necessarie modifiche ai programmi (utilizzo dei soli oggetti). Nella modalità batch l'Input/Output del terminale richiesto dalle applicazioni viene sostituito in modo automatico (necessitano i sorgenti) con chiamate ad API NetAcross. In questa modalità non necessità processore in modalità 5250.
Env: INT=interattivo BCH=Batch.

La struttura degli Html per i pannelli dell'applicazione ha una struttura preordinata (vedere più avanti) ed è sensibile all'applicazione, alle personalizzazioni, alla lingua, ecc. I dati qui inseriti permettono al runtime di determinare il giusto Html nella struttura da servire a questo specifico utente.
Livello autorità: definisce il livello di autorità per accedere alle funzioni estese (vedere Gestione Funzioni Estese in questo manuale). Ogni voce di oggetto funzione estesa ha un livello di autorità espresso da un numero di due cifre. Se il livello di autorità inserito nell'utente è maggiore o uguale di quello inserito nella funzione estesa la funzione è disponibile, altrimenti risulterà disabilitata nel menù.
Utilità Oggetto: definisce l'oggetto di funzione estesa (vedere Gestione Funzioni Estese in questo manuale) che sarà disponibile quando viene richiesto il tasto destro a livello di pagina generale. Corrisponde come funzionalità al tasto Atten dell'iSeries. La funzione estesa a cui ci si riferisce dovrebbe quindi contenere le funzioni di utilizzo comune come la gestione stampe, la gestione PDF, ecc. Viene fornita a tale scopo insieme al prodotto con le funzioni di cui sopra la

funzione speciale Utility.
Utente Gruppo: definisce un gruppo a cui potranno far riferimento le funzioni che possono essere parzializzate per utente (ad es. Generazione Fogli di Calcolo, Funzioni Estese, ecc.). L'utente di gruppo non occorre sia definito come utente reale.
CSS: E' possibile specificare che un utente adotti sempre uno stile di rappresentazione all'interno della pagine Html (colori, font, ecc.). Questo viene attuato assegnando in questo campo un nome di CSS. Se il nome ha un percorso assoluto (inizia per barra) viene ricercato direttamente dentro la directory STYLE. Se ha un percorso relativo segue la ricerca con le regole della applicazione e della libreria (vedere Gestione CSS variabili nel manuale Installazione e Configurazione).
Applicazione: è il nome dell'applicazione a cui questo utente è collegato. Deve corrispondere ad una directory dell'IFS ed ad una istanza dell'Http Server.
Personalizzazione: è il nome di una suddivisione PERS dell'albero delle directory in cui ricercare eventuali personalizzazioni.
Lingua: Indica la lingua in cui fornire le pagine Html. Da una lingua base vengono sostituite a runtime le costanti al fine di presentare una mappa tradotta.

Al momento della conferma della registrazione dell'utente verrà data autorità all'utente QTMHHTP1 di usare l'utente di Logon, per motivi di funzionalità dell'applicazione.

Nel caso di Validation List USRPRF, per ogni utente i dati corrispondenti sono:

Al momento dell'accesso all'utente viene richiesto di autenticarsi. A tal fine viene richiesto in maniera codificata l'inserimento di un utente e di una password.
Utente AS400 (obbligatorio): è il nome utente richiesto al momento della validazione Web. Deve corrispondere ad un user profile iSeries, deve essere quindi lungo al massimo 10 caratteri

ed essere inserito in caratteri maiuscoli. La password richiesta al momento della validazione Web deve essere quella corrispondente all'user profile nel caso di accesso interattivo (vedere più avanti il parametro ENV); nel caso di accesso batch la password non è significativa e può essere inserito qualsiasi valore (eccetto password vuota).
Gli utenti che accedono come utenti di sistema vengono considerati come tipologia ALL e non vengono limitati dagli applicativi se non in base al nome utente.

Per ogni utente collegato viene creata una sessione che esegue il lavoro iSeries. In questo paragrafo vengono specificati i dati per il collegamento che determina l'ambito di operatività (menù).
Si può indicare anche un nome da assegnare, cosa che facilità l'assegnazione di eventuali stampanti o funzioni automatiche.
Logon Terminale: è il nome da usare per la sessione. Se non esiste sarà creato. Se non viene indicato verrà usato un nome nella forma NETVT<progr>.
Logon Menù/Programma: sono i dati per la connessione alla sessione di lavoro.

I dati trasmessi dall'Http Server sono in formato HTML, un formato testo molto discorsivo. Le pagine Html acquistano spesso ampiezze rilevanti, compromettendo quindi la velocità ed i tempi di risposta. Un algoritmo di compressione molto efficiente permette di ridurre tale tempi in maniera drastica.
Compressione GZip: Y=abilitata N=Disabilitata A=Auto (da XML di configurazione). Per utilizzo Intranet locale è preferibile disattivare la compressione dei dati.

Gli iSeries prevedono diversi ambienti di funzionamento relativamente alle modalità cosiddette Interattive e Batch. NetAcross può funzionare in entrambe le modalità, ognuna con diverse considerazioni da tenere presenti. Nella modalità interattiva NetAcross interagisce con l'applicativo intercettando l'Input/Output del terminale e non sono necessarie modifiche ai programmi (utilizzo dei soli oggetti). Nella modalità batch l'Input/Output del terminale richiesto dalle applicazioni viene sostituito in modo automatico (necessitano i sorgenti) con chiamate ad API NetAcross. In questa modalità non necessità processore in modalità 5250.
Env: INT=interattivo BCH=Batch.

La struttura degli Html per i pannelli dell'applicazione ha una struttura preordinata (vedere più avanti) ed è sensibile all'applicazione, alle personalizzazioni, alla lingua, ecc. I dati qui inseriti permettono al runtime di determinare il giusto Html nella struttura da servire a questo specifico utente.
Livello autorità: definisce il livello di autorità per accedere alle funzioni estese (vedere Gestione Funzioni Estese in questo manuale). Ogni voce di oggetto funzione estesa ha un livello di autorità espresso da un numero di due cifre. Se il livello di autorità inserito nell'utente è maggiore o uguale di quello inserito nella funzione estesa la funzione è disponibile, altrimenti risulterà disabilitata nel menù.
Utilità Oggetto: definisce l'oggetto di funzione estesa (vedere Gestione Funzioni Estese in questo manuale) che sarà disponibile quando viene richiesto il tasto destro a livello di pagina generale. Corrisponde come funzionalità al tasto Atten dell'iSeries. La funzione estesa a cui ci si riferisce dovrebbe quindi contenere le funzioni di utilizzo comune come la gestione stampe, la gestione PDF, ecc. Viene fornita a tale scopo insieme al prodotto con le funzioni di cui sopra la

funzione speciale Utility.
Utente Gruppo: definisce un gruppo a cui potranno far riferimento le funzioni che possono essere parzializzate per utente (ad es. Generazione Fogli di Calcolo, Funzioni Estese, ecc.). L'utente di gruppo non occorre sia definito come utente reale.
CSS: E' possibile specificare che un utente adotti sempre uno stile di rappresentazione all'interno della pagine Html (colori, font, ecc.). Questo viene attuato assegnando in questo campo un nome di CSS. Se il nome ha un percorso assoluto (inizia per barra) viene ricercato direttamente dentro la directory STYLE. Se ha un percorso relativo segue la ricerca con le regole della applicazione e della libreria (vedere Gestione CSS variabili nel manuale Installazione e Configurazione).
Applicazione: è il nome dell'applicazione a cui questo utente è collegato. Deve corrispondere ad una directory dell'IFS successiva alla directory /PAGES.
Personalizzazione: è il nome di una suddivisione interna alla cartella /PERS dell'albero delle directory in cui ricercare eventuali personalizzazioni.
Lingua: Indica la lingua in cui fornire le pagine Html. Da una lingua base vengono sostituite a runtime le costanti al fine di presentare una mappa tradotta.

Al momento della conferma della registrazione dell'utente verrà data autorità all'utente QTMHHTP1 di usare l'utente di Logon, per motivi di funzionalità dell'applicazione.

Gestione Certificati (SSL) [2.]

Questa voce permette la gestione dei certificati per attivare la connessione protetta SSL, in particolare dei certificati utente collegati alle Validation List. I certificati possono essere creati per garantire che la connessione Internet venga effettuata veramente con la persona alla quale si intende connettersi. Il server può quindi controllare l'identità degli utenti che accedono e quest'ultimi possono controllare l'identità stessa del sito.
E' possibile adottare tre schemi di sicurezza:

1. Di base. Non viene attivata la protezione SSL. Non c'è autenticazione delle parti e i dati vengono trasmessi in chiaro. La password di connessione viene codificata ma non criptata. In questo caso non occorre nessuna configurazione descritta in questo capitolo.
2. SSL lato server. Viene attivata la protezione SSL e i dati vengono trasmessi criptati (lucchetto sul browser) comprensivi della eventuale password di connessione. Viene autenticato il server ma può collegarsi qualsiasi client. Occorre creare una Certificate Authority e un certificato per il server.
3. SSL completo. Viene attivata la protezione SSL e i dati vengono trasmessi criptati (lucchetto sul browser) comprensivi della eventuale password di connessione. Viene autenticato il server ed ogni client. Occorre creare una Certificate Authority, un certificato per il server e distribuire un certificato per ogni client.

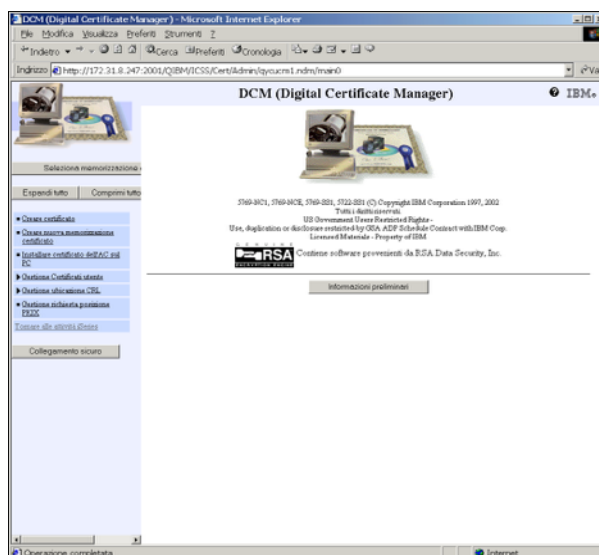
La Certificate Authority ed il certificato server risiedono sull'iSeries.
Il certificato client viene caricato sul browser di chi si deve connettere, contiene l'identità della controparte e sfrutta meccanismi di crittografia per evitarne la contraffazione. Ovviamente la custodia del certificato deve essere effettuata con le dovute regole di riservatezza.
La Certificate Authority privata ed i certificati server e client possono essere creati sull'iSeries.
Prima di selezionare l'applicazione NetAcross Gestione Certificati i certificati devono essere già

stati creati. Il file di configurazione del server Http deve essere stato abilitato per l'SSL (vedere configurazione di esempio). Occorre che sull'iSeries siano stati installati i prodotti di crittografia (vedere Prerequisiti Software nel manuale Installazione e Configurazione) forniti gratuitamente col sistema operativo. La creazione viene poi effettuata avviando l'istanza Admin dell'Http Server,

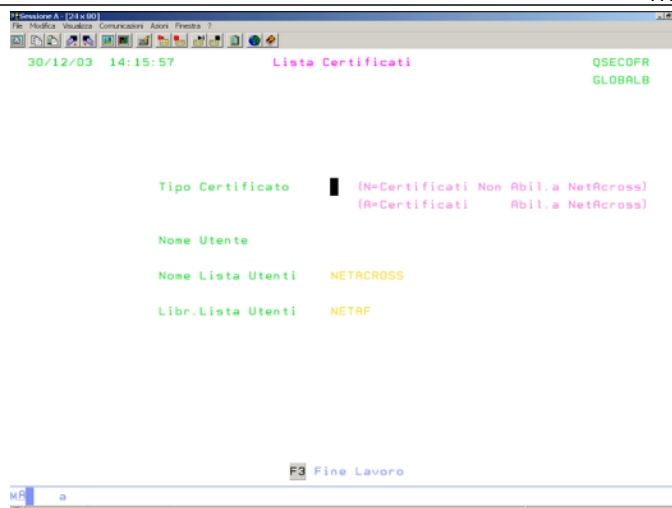
STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)

collegandosi da un browser alla porta 2001,
http:<nome as400>:2001

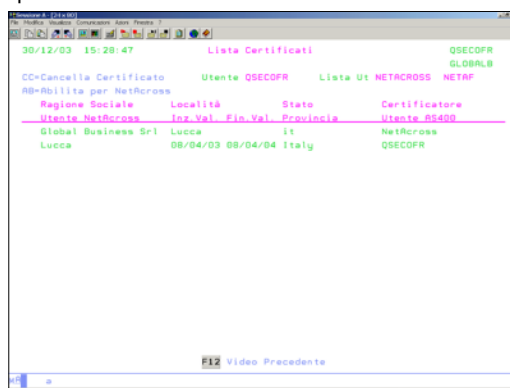
selezionando la sezione Digital Certificate Manager.
Viene richiesto utente e password iSeries. Collegarsi con un utente di livello *SECADM per creare gli oggetti lato server (Certificate authority privata e certificati server).

Appare il seguente schema (per versioni di sistema operativo nuove la visualizzazione può essere diversa:

L'esempio che segue prevede l'utilizzo di una Certificate Authority creata localmente e garantita da noi stessi (sufficiente per un utilizzo B2B). Nel caso di utilizzo al pubblico generico (B2C) è il caso di utilizzare una Certificate Autority garantita pubblicamente. L'esempio che segue in questo caso sarà leggermente diverso.

Creare una Certificate Authority privata. Questo oggetto controlla il rilascio dei certificati effettivi e funziona come sigillo per l'emissione degli oggetti successivi. Selezionare Memorizzazione certificato AC Locale e Crea AC Locale. Riempire i dati richiesti con i dati aziendali. Se si utilizza un certificato del server pubblico questa fase non è necessaria.

Creare un certificato server per l'iSeries. Questo oggetto garantisce l'identità dell'Http Server dell'iSeries. Selezionare Memorizzazione certificato *SYSTEM e Crea Certificato Server. Riempire i dati richiesti con i dati aziendali. Inserire nel Nome Comune il nome del sito da garantire.
Da Applicazioni – Aggiornare Applicazioni Certificato collegare l'applicazione

QIBM_HTTP_SERVER_ + <nome istanza> al certificato creato.
Da Gestione Certificati AC selezionare la Certificate Authority, Abilita e selezionare le applicazioni da garantire (Http Server).
Se si delega la fornitura del certificato server ad una Certificate Authority pubblica occorre dallo stesso menù selezionare la richiesta del certificato, attenderne l'arrivo e poi importarlo nel database.

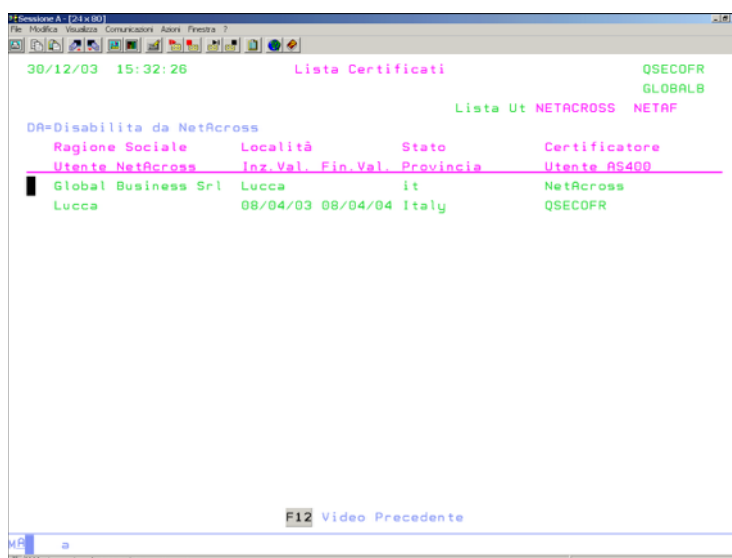
Se si vuole adottare come sicurezza SSL lato server la configurazione è completata e non occorrono ulteriori azioni.
Se si vuole adottare come sicurezza SSL completo si devono creare i certificati per gli utenti (uno per ogni utente). Occorre scollegarsi e ricollegarsi con l'utente specifico che rappresenta il certificato client poiché una volta creati i certificati appartengono all'User Profile che li ha generati. Nella variabile del certificato cliente Common Name occorre inserire un valore che viene confrontato con la chiave <UsrCertName> nel file Config.xml per sicurezza di accesso. Nella variabile del certificato cliente Organization Unit deve essere inserito esattamente (maiuscolo/minuscolo) il nome dell'utente da validare e deve corrispondere ad un utente di sistema o di Validation List a seconda dell'impostazione di configurazione.

Se sono stati generati i certificati client e si vogliono utilizzare in Validation List è possibile assegnarli tramite questa voce di menù. Se invece si vogliono utilizzare con validazione tramite Profilo Utente sono già pronti per l'utilizzo e non è necessaria nessuna funzione per l'attivazione.

Se si vogliono utilizzare i certificati client con validazione tramite Validation List è necessario assegnare i certificati a tale oggetto. Per effettuare questa operazione in modo guidato, attivare

quindi l'opzione di menù Gestione Certificati.

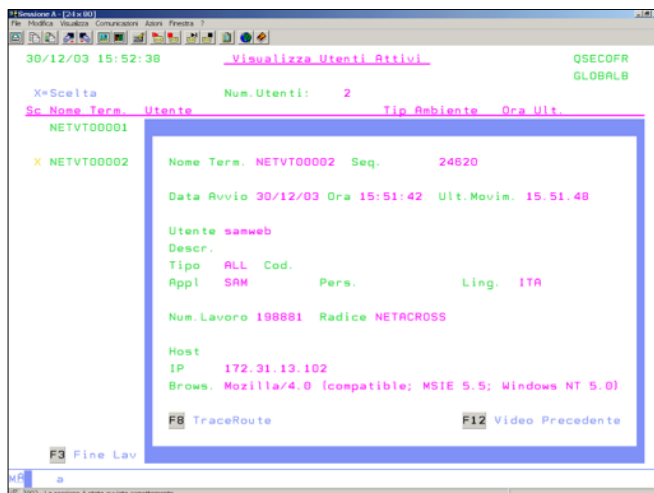
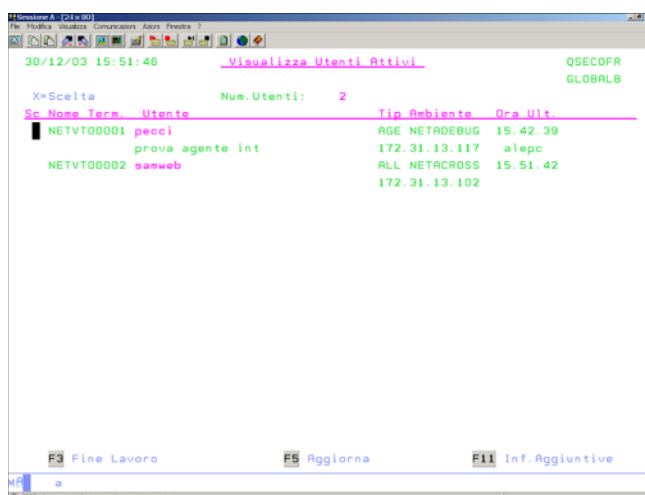
E’ possibile richiedere con l'opzione **N=Certificati Non abilitati a NetAcross** i certificati assegnati ad un determinato profilo utente ma non assegnati ad una Validation List. Questi certificati possono essere usati per l'utilizzo con autenticazione tramite User Profile ma non con le Validation List. Tali certificati possono essere poi assegnati ad una Validation List, che a questo punto è già pronta per l’autenticazione degli utenti via Http Server.
Indicare il nome utente per cui selezionare i certificati e la Validation List a cui verranno assegnati se selezionati nella pagina seguente:

La scelta **CC=Cancella Certificato** permette di cancellare il certificato.
La scelta **AB=Abilita per NetAcross** assegna il certificato alla Validation List indicata permettendone l'utilizzo in modalità protetta (SSL).

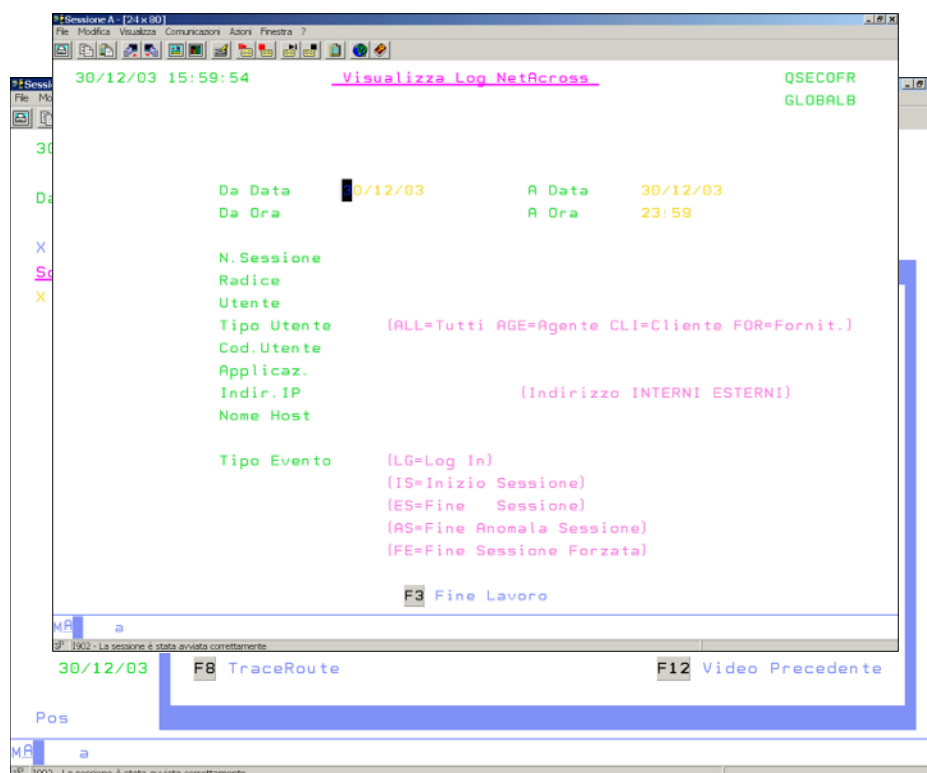
L’opzione inversa (**A=Certificati abilitati a NetAcross**) permette di eliminare un certificato da una Validation List e restituirlo all’utente di proprietà. Come parametri richiede solo il nome della Validation List.

La scelta **DA=Disabilita da NetAcross** disassegna il certificato dalla Validation List indicata riportandolo all'utente proprietario.

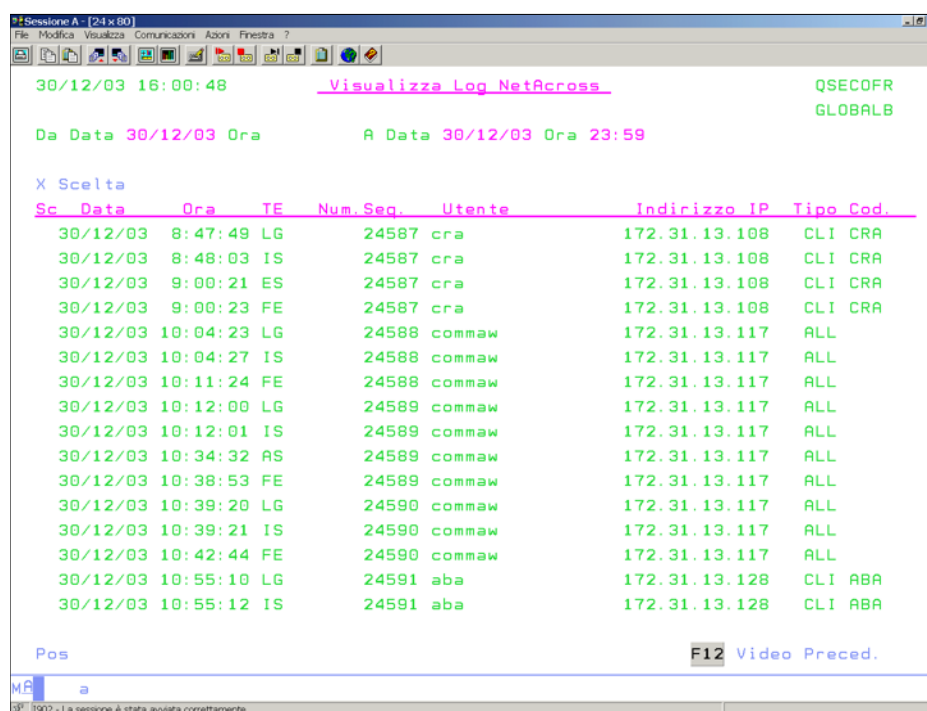
Interrogazione Utenti Attivi [11.]

Gli utenti NetAcross sono un sottoinsieme dei lavori interattivi iSeries ed hanno delle peculiarità (indirizzo IP del client, tempo di inattività, ecc.). Questa opzione permette di visualizzare quali

utenti sono collegati al momento alle applicazioni NetAcross ed interrogarne gli attributi.

La pagina di dettaglio evidenzia i dati particolari:

Da qui è possibile determinare la provenienza dell'accesso se esterno utilizzando la richiesta di TraceRoute.

Interrogazione Log NetAcross [12.]

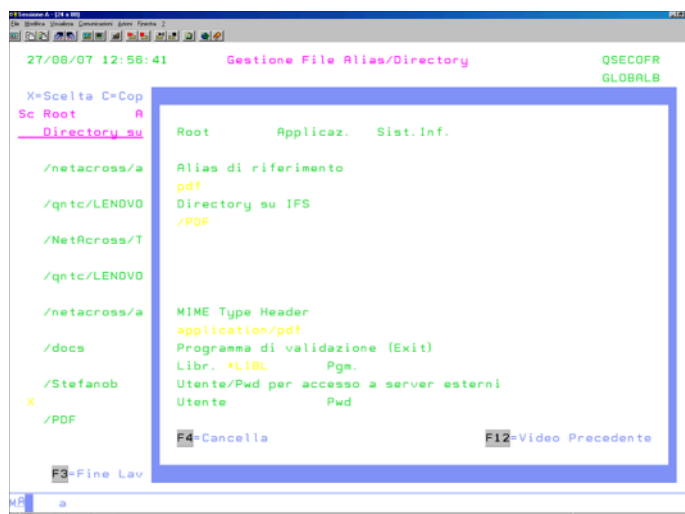
L'Http Server produce in automatico (vedere Configurazione Istanze Http Server nel manuale Installazione e Configurazione) una serie di log di attività riguardanti gli accessi. E' prevista una log per gli oggetti acceduti da parte dei client, degli errori, degli errori CGI. Tali log sono memorizzate nell'IFS dell'iSeries, normalmente nella cartella /<istanza NetAcross</Log.
Il prodotto NetAcross genera un'ulteriore log a livello di database iSeries con una più spiccata vocazione applicativa. In particolare vengono indicati l'inizio e fine sessione, i dati utente, le eventuali terminazioni anomale (dal punto di vista applicativo).
Con questa opzione è possibile l'interrogazione, con ampia possibilità di parzializzazione.

Il dettaglio è come segue:

con l'ulteriore dettaglio:

Anche da qui è possibile selezionare F8=TraceRoute per analizzare in dettaglio la provenienza della connessione.

Gestione Alias/Directory [21.]

La possibilità di sfruttare gli Hyperlink è una funzione fondamentale dell'architettura Web. Gli oggetti a cui puntano gli hyperlink possono essere altre pagine Html, immagini, documenti PDF, ecc. Alcuni di questi oggetti possono essere di carattere riservato (in particolare i documenti PDF se contenenti fatture, ordini, ecc.) e renderli disponibili tramite link significa anche renderli di pubblico dominio.
Questa funzione permette di abilitare la funzione di hyperlink attraverso un server di link controllato dall'applicazione. In particolare si può delegare ad un programma applicativo (vedere Interfaccia per il Programmatore in questo manuale) la decisione se quell'oggetto in base all'utente che lo ha richiesto e al contesto deve essere reso disponibile oppure negato l'accesso. Utilizzando la gestione alias si ha il vantaggio di inserire a livello di Html un nome simbolico, mentre il percorso effettivo degli archivi da collegare è memorizzato sull'iSeries e può essere modificato senza problemi.
L'alias può puntare anche a dati memorizzati su server esterni, sfruttando il supporto Client SMB. A tal fine l'iSeries implementa una directory speciale, di nome QNTC, all'interno della quale sono visibili i PC di classe 2000/XP presenti nella rete all'interno del dominio a cui appartiene l'iSeries. Aprendo i singoli PC si possono vedere le cartelle che questi condividono. L'accesso viene effettuato con la coppia utente e password che in questo caso deve essere fornita in sede di configurazione dell'alias. Tale utente deve essere configurato sia sull'iSeries che sul PC e deve avere la stessa password.
In entrata si può vedere l'elenco degli Alias attualmente inseriti:

entrando nel dettaglio appare la seguente pagina:

Root è il contesto Web a cui si riferisce l'Alias. In particolare è il nome dell'istanza Web.
Applicazione è il nome dell'applicazione a cui si riferisce l'Alias. In particolare viene indicata all'interno dell'utente NetAcross.
Sistema Informativo è il nome del sistema informativo quando ci si riferisce ad applicazioni ACG o SAM. In particolare viene indicato nell'iscrizione utenti di questi due ambienti.
Alias di riferimento è il nome simbolico a cui fa riferimento la pagina Html richiamando un oggetto tramite un link. Utilizzando un nome simbolico permette di poter definire a livello applicativo dove risiedono veramente gli oggetti e di poterli spostare da un server all'altro semplicemente modificando la tabella. Il nome è sensibile al maiuscolo/minuscolo.
Directory su IFS richiede la path del file system integrato iSeries ove cercare l'oggetto.
MIME Type Header è il tipo oggetto che viene comunicato al browser quando gli viene assegnato l'oggetto. Ogni tipologia di immagini, audio, ecc. ha assegnato a livello internazionale una Header Mime. Il nome deve essere compreso nell'elenco permesso dal browser e deve corrispondere al tipo di oggetto contenuto nella directory. Si può lasciare il campo vuoto ed in tale caso viene ricercato il tipo Mime sulla base dell'estensione dell'oggetto ricercato. Vedere la keyword <ObjMime> sul file Config.xml. Alcune estensioni di uso comune (JPG, GIF, ecc.) vengono riconosciute dal browser anche senza bisogno della specifica del tipo Mime.

Libreria e Programma di validazione è il programma iSeries che viene richiamato per ogni oggetto richiesto e che sulla base delle regole esposte in Interfaccia per il Programmatore in questo manuale decide se permettere la visualizzazione dell'oggetto richiesto. Se non viene indicato il programma l'accesso è libero.
Utente e Password per accesso a server esterni è il nome utente e la password di un utente iSeries che viene utilizzato per fornire le credenziali di accesso ad un server esterno. L'utente deve esistere su iSeries e deve corrispondere anche all'utente del server esterno. Se si accede a dati residenti sull'iSeries non è necessario inserire alcun dato.

L'utilizzo dei campi Root, applicazione e sistema informativo permette di differenziare gli alias in

funzione degli ambienti utilizzati. Ad esempio la fattura 125 deve puntare a due directory differenti se la richiesta proviene da due sistemi informativi diversi.
La ricerca dell'Alias avviene col seguente ordine:

Istanza Web	Applicazione	Sist.Informativo
X	X	X
X	X	
X		X
X		
	X	X
	X	
		X

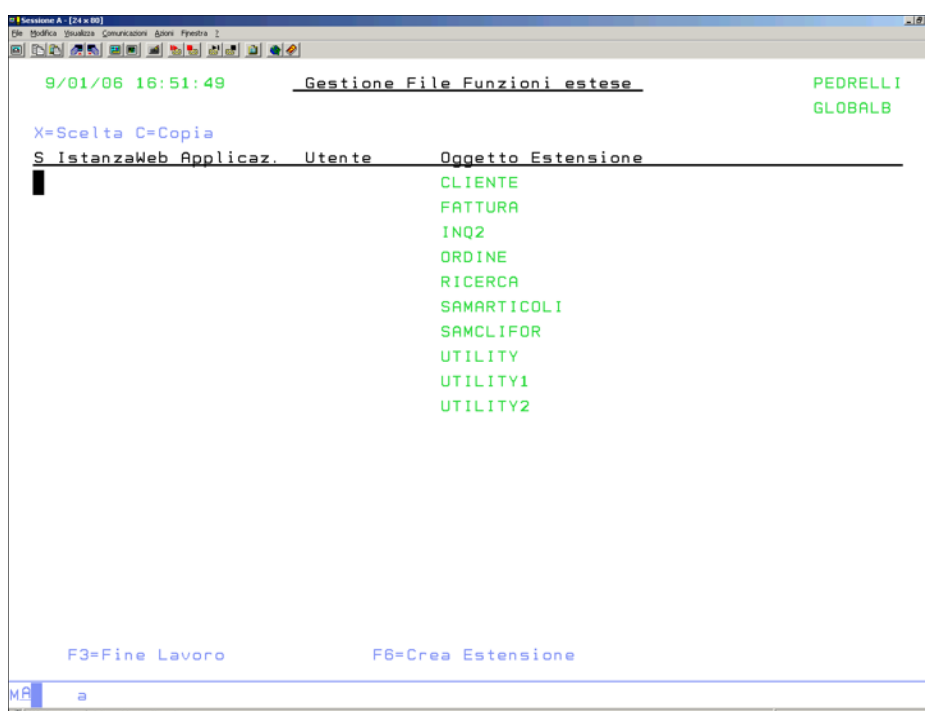
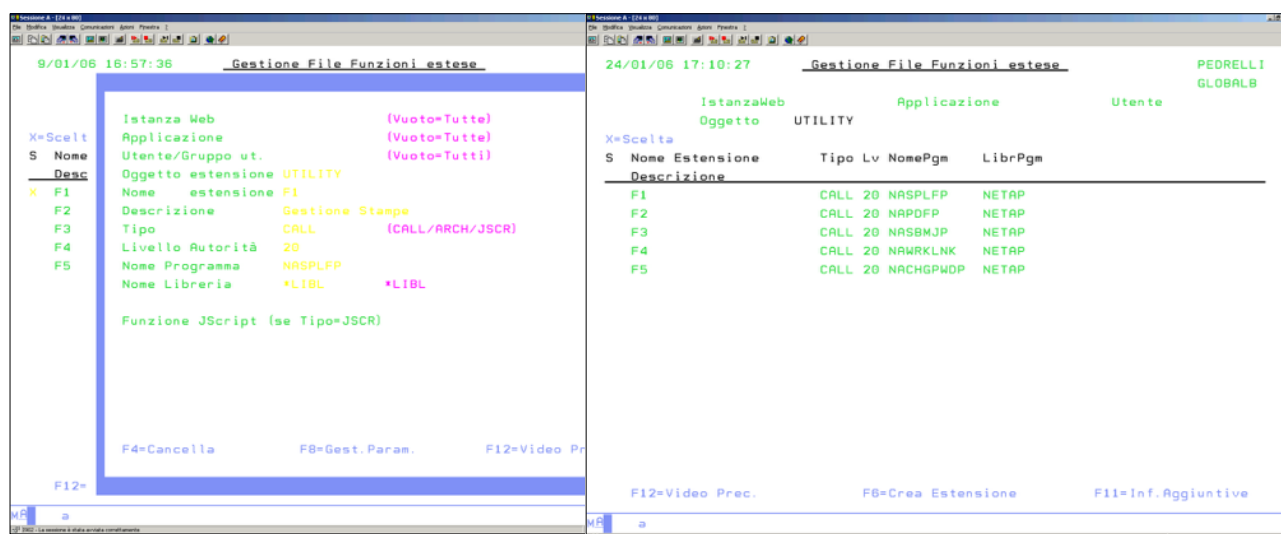
Se si utilizza la visualizzazione dei PDF tramite la Funzione estesa Gestione PDF (vedere paragrafo seguente) deve essere definito un alias come segue:
Alias di riferimento: pdf
Directory su IFS: /PDF
MIME Type Header: application/pdf
Lib.Validazione:
Ut/Pwd per server esterni:

Vengono fornite insieme al prodotto tre alias: nadocs, pdf e xls.

nadocs è l'alias per la Gestione Documenti. Deve puntare ad un indirizzario creato dall'utente nel file system dell'iSeries. Può puntare anche ad un server esterno. Non occorre che abbia un Mime Type Header specifico. Vedere Gestione Documenti nel manuale Installazione e Configurazione.
pdf è l'alias per la presentazione dei documenti PDF. Deve puntare ad un indirizzario creato dall'utente nel file system dell'iSeries. Può puntare anche ad un server esterno. Deve avere un Mime Type Header application/pdf. Vedere Lancio automatico di documenti – PDF nel manuale Installazione e Configurazione per una spiegazione della struttura interna della directory.
xls è l'alias per la presentazione di subfile dentro fogli di calcolo. Deve puntare ad un indirizzario creato dall'utente nel file system dell'iSeries. Deve avere un Mime Type Header application/vnd.ms-excel. Vedere Integrazione con Fogli di Calcolo nel manuale Installazione e Configurazione.
doc è l'alias per la trasformazione delle stampe in formato DOC o PDF. Deve puntare all'indirizzario /<istanza NetAcross>/temp.

Gestione Funzioni Estese [22.]

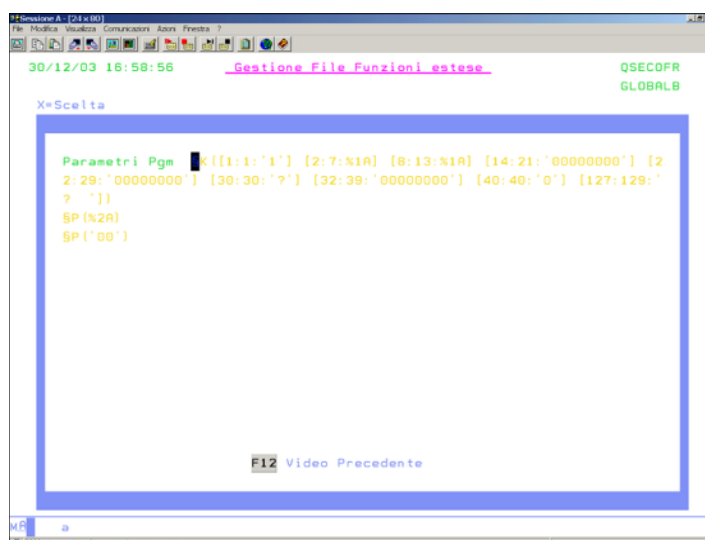
A livello di Html possono essere definite le cosiddette funzioni estese. In particolare è possibile

abilitare un campo ad una funzione estesa ed assegnargli un nome. In questo caso quando il cursore passa sul campo il puntatore si trasforma in un dito. Premendo il tasto destro compare un menù di azioni abilitate dalla funzione estesa. L'elenco delle funzioni corrispondenti ad una funzione estesa e le azioni ad esse corrispondenti vengono inserite con questo programma.
Per fare un esempio si può abilitare una funzione estesa sul codice cliente che richiama le funzioni di estratto conto, elenco ordini, elenco documenti, ecc. Questa funzione permette quindi di espandere la funzionalità del programma originale consentendo di accedere a delle funzioni generalizzate e focalizzate sull'oggetto (ad esempio il cliente).
Uscendo dalla funzionalità normale del programma e richiamando delle funzioni originalmente non accedibili dall'applicazione i singoli programmi a cui accedono le funzioni estese (non quelli corrispondenti alle pagine che le richiamano) devono essere compilate per l'esecuzione in modalità batch NetAcross (vedere capitolo Modalità di Esecuzione nel manuale Installazione e Configurazione).
Esiste poi una funzione estesa speciale che viene attivata quando viene premuto il tasto destro del mouse non in corrispondenza di un campo speciale. Questa è la funzione speciale generica. Tale funzione viene assegnata utente per utente (vedere Gestione Iscrizione Utenti in questo manuale).
Tutte le funzioni sono assegnabili specificatamente a una istanza Web, a una applicazione, a un utente, ad un gruppo utenti o ad una combinazione di esse. La precedenza nell'attribuzione è nell'ordine con cui sono state descritte.
In entrata applicazione viene presentato l'elenco delle funzioni inserite:

Selezionando Copia si può duplicare una funzione estesa. Selezionando Scelta si accede al dettaglio della funzione:
si può selezionare il dettaglio:

Istanza Web è il nome dell'istanza Web (creata con il Wizard di configurazione – vedere Configurazioni Istanze HTTP Server nel manuale Installazione e Configurazione) a cui fa riferimento la funzione. Se non inserito il riferimento è generico.
Applicazione è il nome dell'applicazione (creata con la Gestione Progetti – vedere Gestione Progetti in questo manuale) a cui fa riferimento la funzione. Se non inserita il riferimento è generico.
Utente/Gruppo Utenti è il nome dell'utente (creato con la Iscrizione Utenti – vedere Gestione Iscrizione Utenti in questo manuale) a cui fa riferimento la funzione. Il riferimento può essere fatto ad un singolo utente oppure ad un gruppo (il gruppo viene referenziato all'interno di ogni singolo utente). La precedenza è prima l'utente e poi il gruppo). Se non inserito il riferimento è generico.
Oggetto estensione è il nome delle funzione estesa e deve essere indicato nel file Html. I campi con funzioni estese attivate sono riconoscibili perché il puntatore si trasforma in una mano quando viene posizionato sopra il campo. I nomi che iniziano con INQ sono riservati alle ricerche e se presenti il puntatore viene invece trasformato in un punto interrogativo.
Nome estensione è un nome della singola opzione della funzione. Per una funzione estesa possono essere richiamate più funzioni applicative indicando più nomi estensione.
Descrizione è un nome descrittivo della funzione e viene riportato nel menù contestuale quando viene premuto il tasto destro del mouse.
Tipo è il tipo di applicazione da eseguire. E' possibile inserire:

CALL per la chiamata ad un oggetto di tipo *PGM.

JSCR per richiamare una funzione JavaScript.

Livello autorità è un numero per determinare l'accessibilità della funzione e viene confrontato con l'omologo campo della gestione utenti. Il livello autorità dell'utente deve essere uguale o superiore al livello di autorità della funzione per permetterne l'utilizzo.
Nome Programma e Libreria indicano il programma da richiamare per la funzione estesa se indicato CALL nel tipo funzione. Tale programma deve essere compilato per l'esecuzione in modalità batch NetAcross. I parametri della Call vengono definiti con F8=Gest.Parametri.
Funzione JScript prevede l'inserimento del codice JavaScript da eseguire se indicato JSCR nel tipo funzione.

Se premuto F8=Gest.Parametri appare una pagina per l'inserimento dei parametri della Call:

L'inserimento dei parametri richiede una sintassi specifica:

i parametri del programma iSeries sono mappati secondo l'ordine in cui sono espressi dalla sintassi. L'inizio di un parametro è segnato dal carattere §.
si avrà:
§P(per indicare un parametro di CALL
§K(per indicare che questo parametro è la KPJBU del modulo base ACG

la fine del parametro sarà la parentesi tonda chiusa :)
per costruire la KPJBU o i parametri Call è possibile inserirli come unica stringa o come somma di elementi posizionali secondo questa sintassi:
[*carattere_iniziale* : *carattere_finale* : *valore_da_impostare*] [*car_i* : *car_f* : *value*]....
esempio:
[01:10:'ARTICOLO00']

Il *carattere_iniziale* è un valore numerico da 1 a 3 cifre
Il *carattere_finale* è un valore numerico da 1 a 3 cifre
Il *valore_da_impostare* potrà essere:

' **valore** ' (*apice* valore *apice*) valore carattere impostato fisso nel parametro
valore (valore) valore numerico impostato fisso nel parametro
%numeroA in cui la %numero indica l'ennesimo parametro della na_ext inteso come ALFABETICO
%numeroN in cui la %numero indica l'ennesimo parametro della na_ext inteso come NUMERICO

Esempio di parametri:
Il primo parametro è la KPJBU con una serie di valori impostati, alcuni fissi, alcuni riferiti al primo parametro dell'html inteso alfabetico.
Il secondo parametro è il secondo parametro dell'html passato alfabetico
Il terzo parametro è il valore alfabetico fisso '00'
§K([1:1:'1'] [2:7:%1A] [8:13:%1A] [14:21:'00000000'] [22:29:'00000000'] [30:30:'?']

[32:39:'00000000'] [40:40:'0'] [127:129:'? '])
§P(%2A)
§P('00')

Vengono fornite insieme al prodotto cinque funzioni di utilità: la gestione stampe, la gestione PDF, la gestione lavori immessi, la gestione documenti ed il cambio password. Nell'estensione UTILITY vengono incluse tutte e cinque (intesa quindi per utenti Intranet); nell'estensione UTILITY1 vengono incluse Gestione PDF, Gestione Documenti e Cambio Password (intesa quindi per utenti Internet).

La gestione stampe

Prevede le funzioni comunemente usate della gestione spool del sistema operativo (WrkSplF). È definita come segue:

```
Oggetto estensione: UTILITY
Nome estensione: F1
Descrizione: Gestione Stampe
Tipo: CALL
Livello autorità: 20
Libreria/Programma: NETAP/NASPLFP
Parametri Pgm: §P('U')
```

La gestione stampe visualizza gli spool relativi all'utente che li ha generati. E' possibile specificare un parametro che indica se visualizzare le stampe di registrazione errori del sistema (Joblog, Debug) o no. Il parametro di un carattere ha i seguenti significati:
U livello utente: non visualizzare le stampe delle registrazioni errori. Può vedere solo le proprie stampe.
E livello esperto: visualizzare le stampe delle registrazioni errori. Se autorizzato può vedere le stampe degli altri utenti.
Se omesso vale E. Per registrazioni errori si intendono le stampe inserite nelle code QEZJOBLOG o QDEBUG nella libreria QUSRSYS.
Nel caso di utenti collegati tramite Validation List più utenti utilizzano lo stesso utente di sistema di collegamento. In questo caso l'utilità di gestione spool NON andrebbe abilitata.
Questa procedura è inseribile anche nei menù utente (Call NetAP/NASPLFP Parm(U/E)).

La gestione PDF

Prevede la gestione dei file PDF generati sul file system iSeries, normalmente tramite il prodotto Infoprint Server (vedere Prerequisiti Software nel manuale Installazione e Configurazione). Tali PDF devono risiedere nella directory referenziata dall'alias "pdf" (vedere Gestione Alias/Directory in questo manuale). Al suo interno deve esistere una cartella col nome utente di sistema collegato ed al suo interno devono essere contenuti i files PDF col nome standard attribuito dal prodotto Infoprint Server, e cioè
<num.job>_<num.spl>_<nome.job>_<data.job(mmggaaaa)>_<num.seq>.PDF.
Num.job è il numero del lavoro; può essere lungo da 1 a 6 caratteri.

Num.spl è il numero dello spool; può essere lungo da 1 a 6 caratteri.
Nome.spl è il nome dello spool; può essere lungo da 1 a 10 caratteri.
Data job è la data di creazione dello spool; deve essere lungo 8 caratteri (mmggaaaa).
Num.seq è il numero sequenza dello spool; può essere lungo da 1 a 6 caratteri.
La funzione è definita come segue:
Oggetto estensione: **UTILITY**
Nome estensione: **F2**
Descrizione: Gestione **PDF**
Tipo: **CALL**
Livello autorità: **20**
Libreria/Programma: **NETAP/NAPDFP**

La gestione PDF visualizza i files PDF relativi all'utente che li ha generati. Nel caso di utenti di sistema operativo tutto funziona come previsto. Nel caso di utenti collegati tramite Validation List più utenti utilizzano lo stesso utente di sistema di collegamento. In questo caso occorre attribuire i singoli files specificamente all'utente di Validation List. Per fare questo occorre modificare il programma che crea il file di spool modificando il nome spool (parametro SPLFNAME del comando OVRPRTF) in NA<n.sequenza>. Il numero di sequenza si ottiene chiamando il programma NetAcross NETAINFSP con due parametri:
Esito, di 1 carattere alfabetico.
N.Sessione, di 10 caratteri alfabetico.
Entrambi i parametri vengono inizializzati dal programma, per cui non importa con quali valori vengono forniti.
I valori di ritorno valgono come segue:
Esito = E. Errore, il lavoro non è stato riconosciuto come lavoro NetAcross. N.Sessione vale zero.
Esito = S. Il lavoro è riconosciuto come lavoro NetAcross, però l'utente non è un utente di Validation List ma di sistema, per cui non occorre la modifica del nome spool. N.Sessione vale zero.
Esito = spazio. Il lavoro è riconosciuto come lavoro NetAcross, l'utente è un utente di Validation List. Il N.Sessione è il valore da inserire nel parametro SPLFNAME per sostituire il nome Spool da creare. Il nome è già comprensivo del prefisso NA.

Questa procedura è inseribile anche nei menù utente (Call NetAP/NAPDFP).

La gestione lavori immessi

Prevede le funzioni comunemente usate della gestione lavori del sistema operativo (WrkSbmJob).
È definita come segue:
Oggetto estensione: **UTILITY**
Nome estensione: **F3**
Descrizione: **Gestione Lavori Immessi**
Tipo: **CALL**
Livello autorità: **20**
Libreria/Programma: **NETAP/NASBMJP**
Parametri Pgm: **§P('U')**

La gestione lavori immessi visualizza i lavori immessi relativi all'utente che li ha generati. Il parametro di un carattere ha i seguenti significati:

U livello utente: se richiesto di visualizzare le stampe non visualizzare quelle relative alle registrazioni errori.
E livello esperto: se richiesto di visualizzare le stampe visualizzare anche quelle relative alle registrazioni errori.
Se omesso vale E. Per registrazioni errori si intendono le stampe inserite nelle code QEZJOBLOG o QDEBUG nella libreria QUSRSYS.
Questa procedura è inseribile anche nei menù utente (Call NetAP/NASBMJP Parm(U/E)). Nel caso di utenti collegati tramite Validation List più utenti utilizzano lo stesso utente di sistema di collegamento. In questo caso l'utilità di gestione lavori immessi NON andrebbe abilitata.

La gestione documenti

Prevede la distribuzione agli utenti del contenuto di una cartella in formato PC in modo protetto (in particolare per il Web). È definita come segue:

`Oggetto estensione:` **`UTILITY`**
`Nome estensione:` **`F4`**
`Descrizione:` **`Gestione Documenti`**
`Tipo:` **`CALL`**
`Livello autorità:` **`20`**
`Libreria/Programma:` **`NETAP/NAWRKLNK`**

La gestione documenti ricerca i documenti a partire dall'alias che fa riferimento alla cartella principale indicata nel file Config.xml con la keyword MainDocAlias.

<MainDocAlias>*nome_alias*<MainDocAlias>

Nella gestione Alias del meno CONF è necessario definire l’alias di cui sopra che punti alla cartella scelta come radice principale delle sottocartelle per utente.

All'interno di tale cartella infatti, viene ricercata una cartella con il nome dell'utente collegato. Gli utenti vedranno i documenti ivi contenuti ed avranno la possibilità di visualizzarli, salvarli, stamparli e cancellarli. E' possibile gestire oltre ai link fisici anche i link simbolici, permettendo così di condividere delle cartelle (ad esempio listini comuni o immagini) a più utenti.

Se l'utente ha l'autorizzazione in modifica alla cartella è possibile caricare degli archivi dal PC dell'utente alla sua cartella documenti. In questo caso viene abilitato un pulsante che permette di indicare il file da trasferire (con la possibilità di ricerca). Per motivi di sicurezza è possibile limitare l'ampiezza massima del singolo file che è possibile trasferire (vedere Esempio di Configurazione Http nel manuale Installazione e Configurazione).

Può essere utile avere un'area comune, per questo è necessario creare la cartella /root/comune linkare in ogni sottocartella utente la cartella comune.
Se la cartella principale è /docs è possibile linkare una cartella generica a tutti impostando per ogni utente:

MKDIR /docroot/comune
ADDLNK OBJ('/docs/comune') NEWLNK(‘/docs/*utente*/comune') /* *per ogni utente* */

Riferimenti in configurazione

L'area <PasswordList> permette di definire una regola di scadenza password per ogni validation list indicando con 3 cifre numeriche i giorni di validità di una password <GgPwdValid> , i giorni di preavviso <GgPwdAlert> e il numero minimo di caratteri <GgPwdMinCh>.
Una volta impostate le regole per la validation list scelta il controllo password è attivato.

Nel caso di utenti di sistema USRPRF/NETAF l'unico valore utilizzato è quello dei giorni di preavviso e il controllo si attiva solo se l'utente ha la password a scadenza impostata sul profilo.

```
<PasswordList>
        <Password>
                <PasswordName>NETACROSS/NETAF</PasswordName>
                <GgPwdValid>090</GgPwdValid>
                <GgPwdAlert>030</GgPwdAlert>
                <GgPwdMinCh>006</GgPwdMinCh>
        </Password>
</PasswordList>
```

Nel caso di Authentication Basic , il controllo per password scaduta, sugli utenti di sistema non è attivabile in quanto è Apache stesso che impedisce l'accesso, quindi l'utente deve essere attivato da un amministratore

Il cambio password

Permette di cambiare la password dell'utente. È definita come segue:
Oggetto estensione: **UTILITY**
Nome estensione: **F5**
Descrizione: **Cambio Password**
Tipo: **CALL**
Livello autorità: **20**
Libreria/Programma: **NETAP/NACHGPWDP**

Vengono cambiate le password dell'utente (Validation List) e se utente di sistema viene allineata la password anche sull'User Profile. Viene richiesta la vecchia password per conferma.

La gestione password scaduta

Per permettere ad un utente di cambiare la propria password quando questa è scaduta è necessario creare una particolare funzione estesa definita in questo modo:
Oggetto estensione: **PASSWORD**
Nome estensione: **PWD**
Descrizione: **Cambio Password**
Tipo: **CALL**
Livello autorità: **20**
Libreria/Programma: **NETAP/NACHGPWDP**

Questa funzione non sarà visibile in nessun menù ma verrà attivata dal sistema quando una

password sta per scadere o è scaduta sia per utenti di validation list che per utenti di sistema in base alla configurazione sul file config.xml

La gestione stampanti

Prevede le funzioni comunemente usate della gestione delle stampanti del sistema operativo (WrkWtr). È definita come segue:

```
Oggetto estensione: UTILITY
Nome estensione: F6
Descrizione: Gestione Stampanti
Tipo: CALL
Livello autorità: 20
Libreria/Programma: NETAP/NAWRKWTRP
Parametri Pgm: §P('U')
```

La gestione stampanti visualizza i lavori di scrittura relativi alla stampante associata all'utente che sta usando la funzione. Il parametro di un carattere ha i seguenti significati:
U livello utente: se richiesto di visualizzare le stampe non visualizzare quelle relative alle registrazioni errori. Può vedere solo la propria stampante.
E livello esperto: se richiesto di visualizzare le stampe visualizzare anche quelle relative alle registrazioni errori. Può vedere se autorizzato anche le altre stampanti e gestirne lo stato ed il contenuto.
Se omesso vale E. Per registrazioni errori si intendono le stampe inserite nelle code QEZJOBLOG o QDEBUG nella libreria QUSRSYS.
Questa procedura è inseribile anche nei menù utente (Call NetAP/NAWRKWTRP Parm(U/E)). Nel caso di utenti collegati tramite Validation List più utenti utilizzano lo stesso utente di sistema di collegamento. In questo caso l'utilità di gestione stampanti NON andrebbe abilitata.

La gestione messaggi

Prevede le funzioni comunemente usate della gestione dei messaggi del sistema operativo (DSPMSG e SNDMSG). È definita come segue:

```
Oggetto estensione: UTILITY
Nome estensione: F7
Descrizione: Gestione Messaggi
Tipo: CALL
Livello autorità: 20
Libreria/Programma: NETAP/NADSPMSGP
Parametri Pgm: §P('N')
```

La gestione messaggi visualizza i messaggi relativi al terminale ed all'utente che sta usando la funzione. Il parametro di un carattere ha i seguenti significati:
S vengono visualizzati sia i messaggi di terminale sia quelli dell'utente.
N vengono visualizzati i messaggi dell'utente.
All'interno della funzione è possibile rispondere, cancellare messaggi ed inviarne di nuovi.
Se il lavoro è in attesa di messaggi inviati alla coda dell'operatore di sistema è possibile gestire anche questi messaggi.
Questa procedura è inseribile anche nei menù utente (Call NetAP/NADSPMSGP Parm(S/N)). Nel caso di utenti collegati tramite Validation List più utenti utilizzano lo stesso utente di sistema di

collegamento. In questo caso l'utilità di gestione messaggi NON andrebbe abilitata.

La gestione messaggi tramite Polling

Permette di controllare periodicamente ed automaticamente l'esistenza di messaggi nella propria code del terminale e dell'utente. Se questa funzione è abilitata si viene notificati della presenza di nuovi messaggi e viene proposta la possibilità di vederli ed eventualmente rispondere.
Se si vuole utilizzare questa funzione occorre definire la keyword <UsrMsgPolling> in Config.sys e definire la relativa funzione estesa. È definita come segue:

```
Oggetto estensione: MESSAGE
Nome estensione: MSG
Descrizione: Gestione Messaggi
Tipo: CALL
Livello autorità: 20
Libreria/Programma: NETAP/NADSPMSGWP
Parametri Pgm: nessuno
```

Per avere il pulsante disponibile è necessario inserirlo nella toolbar, quindi inserire nella top.htm se non presente un bottone nella <div> della toolbar.

```
<button class="natool_button" id="natoolkey_MSG" nastatic="true">
 <div class="natool_image" >
 </div>
 <div class="natool_text">
Messaggi
 </div>
</button>
```

Il foglio di stile natoolbar.css associato deve contenere la definizione grafica del bottone, che per esempio potrebbe essere questa:

```
button#natoolkey_MSG div.natool_image {
     background-image: url(tools/24/msg.png);
   background-repeat: no-repeat;
}
```

Riordino delle colonne del subfile esteso

Permette di abilitare un pulsante nella barra delle funzioni del subfile esteso, vicino ai pulsanti di navigazione e di estrazione xls, in modo che gli utenti abilitati possano vedere il subfile con le colonne posizionate secondo preferenza personale.
Sarà attivata solamente se nel config è presente la keyword:

```
<GestUsrNasfo>Y</GestUsrNasfo>
```

```
Oggetto estensione: SUBFILEST
Nome estensione: NASFO
Descrizione: Personalizzazione subfile
Tipo: CALL
Livello autorità: 20
Libreria/Programma: NETAP/NASFLESTP
Parametri Pgm: nessuno
```

Creazione di un template da subfile esteso

E' possible fare in modo che gli utenti si creino i loro template inserendo nella toolbar un pulsante per eseguire l'operazione. La funzione estesa associata è questa:

```
Oggetto estensione: NAXLTCRT
Nome estensione: NAXLTCRT
Descrizione: Creazione template XLS
Tipo: CALL
Livello autorità: 20
Libreria/Programma: NETAP/NAXLTCRTP
Parametri Pgm: nessuno
```

Nella toolbar deve esistere un bottone così definito:

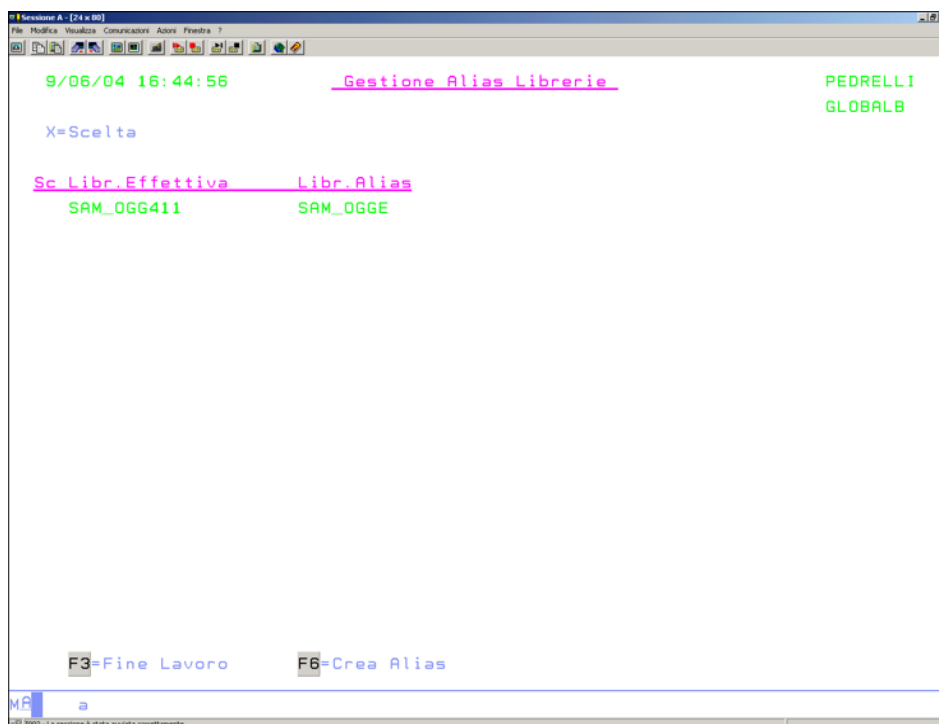
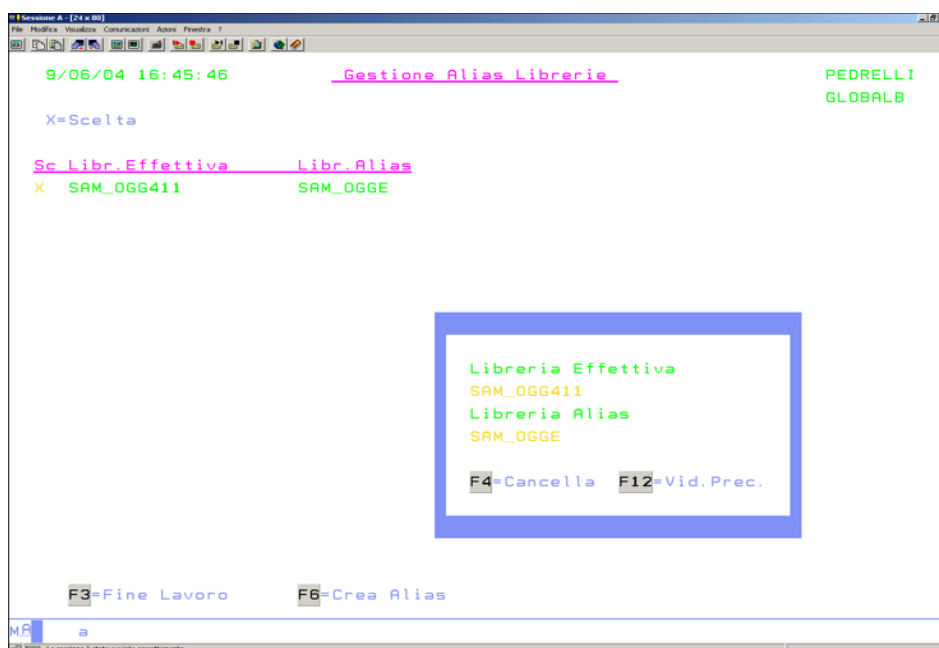
```
<button class="natool_button" id="natoolkey_XLT" nastatic="true">
 <div class="natool_image" >
 </div>
 <div class="natool_text">
Template
 </div>
</button>
```

Nel foglio di stile associato natoolbar.css deve esistere:

```
button#natoolkey_XLT div.natool_image {
     background-image: url(/images/tools/Excel-template.png);
     background-repeat: no-repeat;
}
```

Gestione Alias Librerie [23.]

L'applicazione genera una serie di oggetti a partire dall'oggetto file video che viene utilizzato dai programmi 5250. In particolare viene generato una pagina Html per ogni gruppo di emissione video e vengono registrate le informazioni relative anche negli archivi NADDS* all'interno della libreria NETAF. Se sono richieste le traduzioni viene generato anche un file di testo contenente le parole da tradurre.
Tutte queste informazioni vengono legate all'oggetto file video ed alla libreria che lo contiene.
Ci sono situazioni (ad esempio nella distribuzione di prodotti standard) ove il nome della libreria può essere variato, ma non si vogliono cambiare anche i relativi files NetAcross. In questo caso si può specificare un Alias di libreria, dove si comunica che a fronte di una libreria reale negli archivi NetAcross tale nome va ricercato con un nome diverso.
Questa funzione permette di inserire questi dati. Normalmente non si dovrebbe aver bisogno di questa funzionalità.

In entrata si può vedere l'elenco degli Alias attualmente inseriti:

entrando nel dettaglio appare la seguente pagina:

Libreria Effettiva è il nome della libreria così come si chiama a livello di iSeries.
Libreria Alias è il nome da ricercare per la libreria negli archivi NetAcross.

Se si definisce un alias per una libreria verrà disabilitata automaticamente la funzione di conversione Html “al volo”.

Creazione CSS da CSS base [24.]

L'applicazione prevede di utilizzare CSS diversi in funzione della risoluzione video e dell'ampiezza della finestra del browser. Questa voce permette di manutenere un unico CSS e poi generare quelli ad altre risoluzione in funzione di parametri automatici.
Per dettagli sull'argomento fare riferimento al capitolo in questo manuale Gestione CSS variabili.
Una volta creato un CSS base con le ampiezze ed i colori corretti eseguire questa voce di menù. Apparirà il seguente pannello.

Root è il nome dell'istanza NetAcross che contiene il CSS.
Path Css da copiare è la path interna all'istanza NetAcross riferita alla directory che contiene il CSS da copiare. Se non inserita vale <istanza>Style. Se inserita con la barra in prima posizione parte da <istanza>, altrimenti parte da <istanza>Style.
Path Css da creare è la path interna all'istanza NetAcross riferita alla directory che contiene il CSS da creare. Se non inserita vale <istanza>Style. Se inserita con la barra in prima posizione parte da <istanza>, altrimenti parte da <istanza>Style.

net-Across Creazione Css

Root netacross

Path Css da copiare

Path Css da creare

Nome Css da Copiare netacross

Suffisso per CSS _801240

font-size 2 px

line-height 4 px

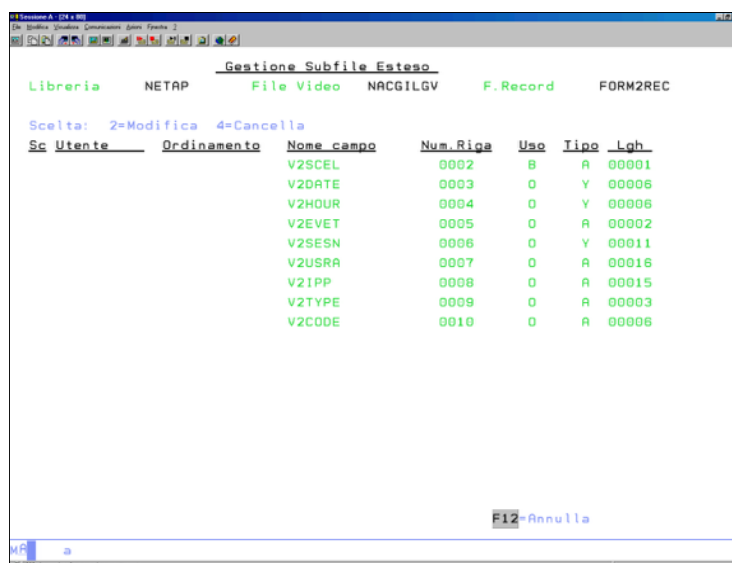
F6 Conferma creazione F12 Uscita

Nome Css da copiare è il nome interno all'istanza NetAcross più la path di cui sopra che si riferisce al CSS da copiare. Il nome deve essere inserito senza il suffisso (.CSS).
Suffisso per CSS è il suffisso che verrà aggiunto al nome del CSS originale. A tale nome si dovrà poi fare riferimento nel file di configurazione all'interno delle keyword DynamicCss.
Font-size è un valore relativo, positivo o negativo, espresso in pixel per l'espansione o riduzione dei valori di dimensione del CSS (keyword font-size).
Line-height è un valore relativo, positivo o negativo, espresso in pixel per l'espansione o riduzione dei valori di dimensione del CSS (keyword line-height).

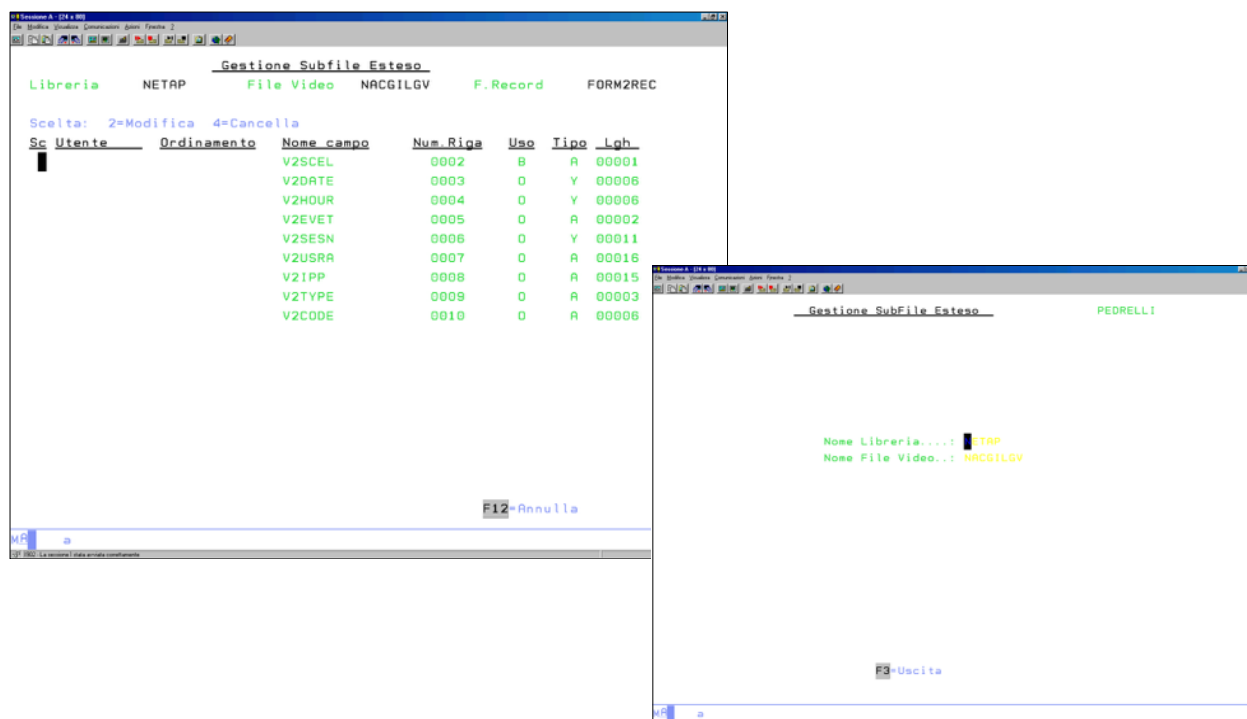
Per semplificare la creazione automatica di una serie di CSS è possibile richiamare il programma direttamente da un programma fatto dall'utente seguendo le seguenti specifiche:

Nome programma: NACRTCSSL.

Parametri (la descrizione è elencata qui sopra):

```
C    *Entry      PList
C               Parm           VaRoot     10
C               Parm           DaCss    256
C               Parm           AdCss    256
C               Parm           Name                      50
C               Parm           Sufix      50
C               Parm           Fontpx     2  0
C               Parm           Linepx     2  0
```

Viene comunque fornito un programma di esempio di nome NADUPCSSP il cui sorgente è nel file QCLLESRC nella libreria NETAP. Tale esempio ricrea i CSS nei vari formati sulla base dei valori di base forniti con la configurazione di default. Se tale programma viene personalizzato deve essere duplicato in librerie utente per evitare che venga sovrapposto in sede di aggiornamento di release.

Gestione Subfile Esteso [25.]

I subfile sono delle strutture che bene si prestano ad estensioni funzionali. NetAcross permette tramite una apposita keyword (vedere Direttive HTML nelle DDS in questo manuale) di ampliare la funzionalità del subfile estendendolo verticalmente ed orizzontalmente. Orizzontalmente appunto viene dataq la possibilità di scegliere quali campi visualizzare ed in quale ordine. Tale scelta può essere effettuata su base generica oppure a livello di utente.
Questa voce di menù permette di effettuare queste scelte.

Una volta selezionato il file video ed il formato record viene evidenziato l'elenco dei campi che lo compongono:

Gestione Subfile Esteso

Libreria NETAP File Video NACGILGV F.Record FORM2REC

Scelta: 2=Modifica 4=Cancella

Sc	Utente	Ordinamento	Nome campo	Num.Riga	Uso	Tipo	Lgh
			V2SCEL	0002	B	A	00001
			V2DATE	0003	O	Y	00006
			V2HOUR	0004	O	Y	00006
			V2EVET	0005	O	A	00002
			V2SESN	0006	O	Y	00011
			V2USRA	0007	O	A	00016
			V2IPP	0008	O	A	00015
			V2TYPE	0009	O	A	00003
			V2CODE	0010	O	A	00006

F12=Annulla

Per ogni campo dell'elenco è possibile indicare se deve essere inserito nel subfile esteso ed in quale ordine. Le opzioni sono inseribili senza utente ed in questo caso sono generiche, oppure sono inseribili con l'utente ed in questo caso valgono solo per quest'ultimo.
Questa funzione è raggiungibile direttamente dal subfile stesso a tempo di esecuzione, tramite l'apposito pulsante.

Gestione Autorizzazioni [26.]

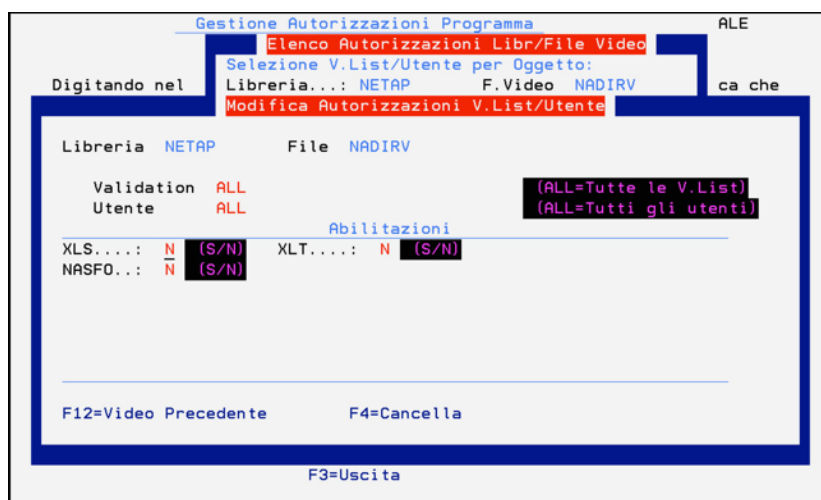
E' possibile impostare le autorizzazioni all'uso dei subfile estesi presenti nei display file, si può stabilire se un utente potrà richiamare l'estrazione del foglio di excel , la creazione dei template e il riordino dei campi del subfile esteso.

Gestione Autorizzazioni Programma ALE

Digitando nel nome della Libreria la parola speciale ALL significa che si vuol autorizzare l'utente,che verrà iscritto successivamente,a tutte le funzioni di tutti i file video di tutte le librerie

Nome Libreria....: NETAP
Nome File Video..: NADIRV

F3=Uscita

Per ogni display si stabilirà le autorizzazioni per gli utenti delle rispettive validation list
Il valore speciale **ALL** per validation list e utente imposta la regola generale per tutti gli utenti.
I valori di autorizzazione sono:

XLS: S / N abilitazione al download del subfile in formato XLS

XLT: S / N abilitazione alla creazione del template. relativo al subfile

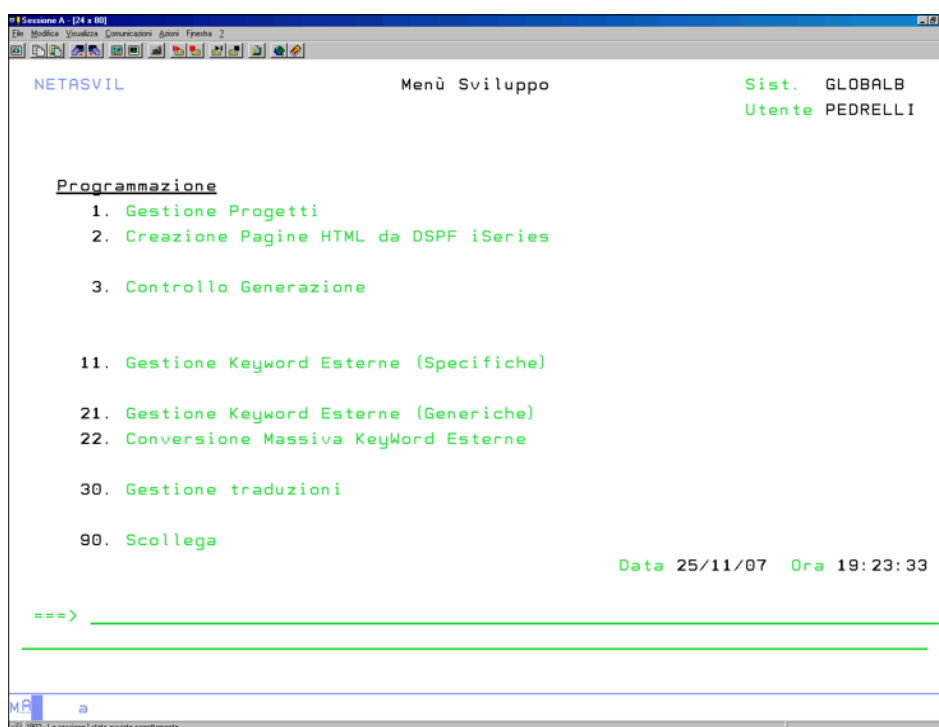
NASFO: S / N abilitazione al riordino delle colonne del subfile

Riferimenti in configurazione

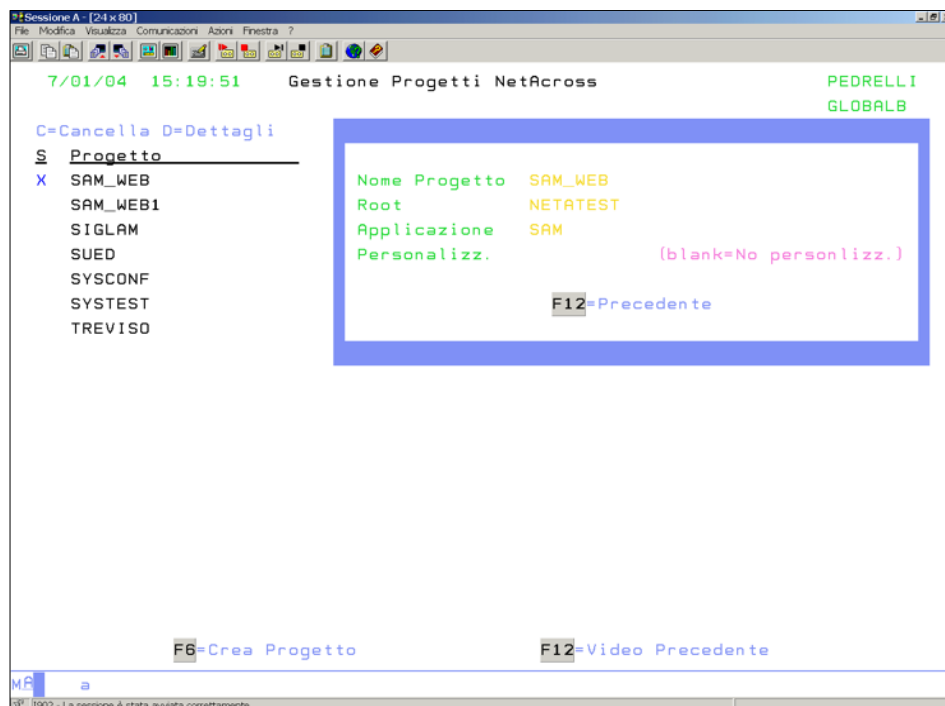
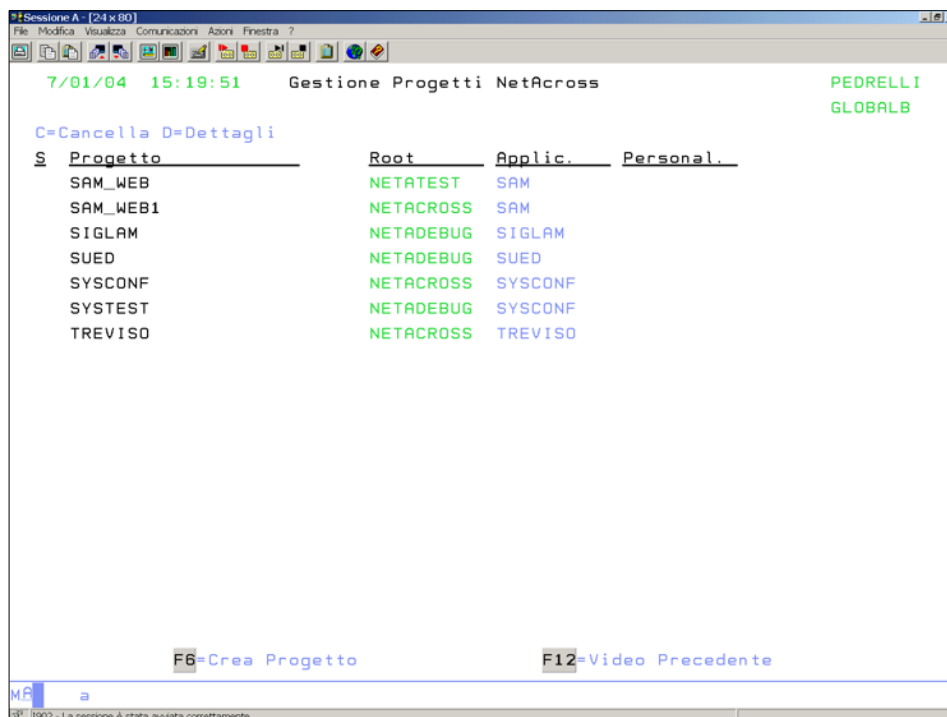
config.xml

```
<GestAuth>S</GestAuth> : Impostando a "S" questo valore si abilitano le
autorizzazioni previste per i subfile estesi. [S/N]

<GestUsrNasfo>S</GestUsrNasfo> : Impostando a "S" questo valore si
attiva la gestione del subfile esteso con le colonne riordinabili come
regola generale.
```

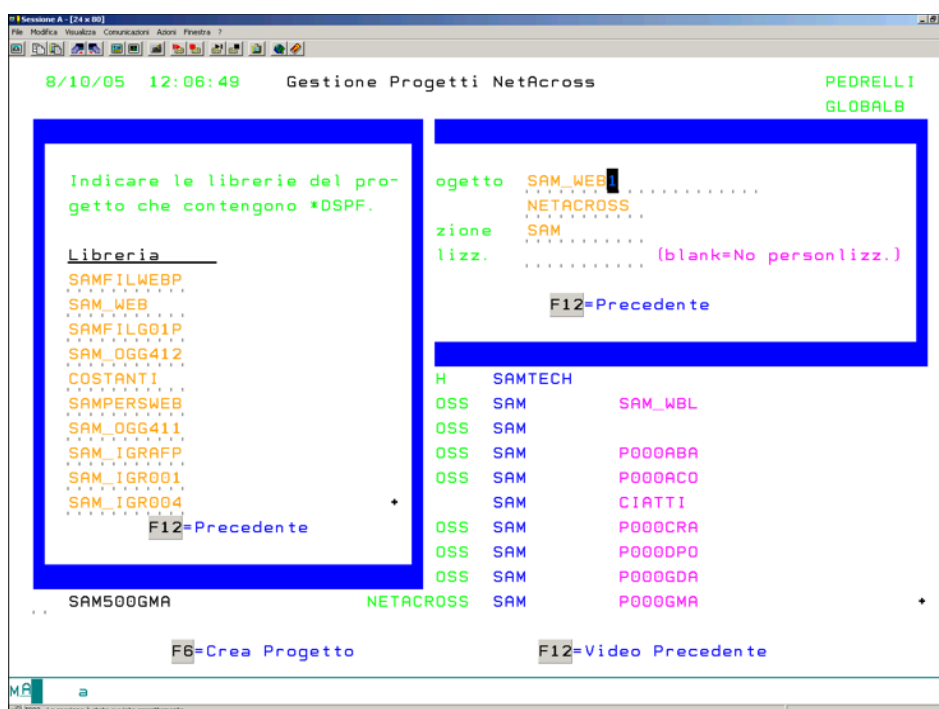
Funzione SVIL

Da questo menù è possibile generare le pagine Html e gli archivi necessari al runtime, creare dei cross-reference per l'avanzamento della creazione degli Html, interrogare le log di conversione.

Gestione Progetti [1.]

Questa funzione permette di creare i progetti, cioè dei nomi che all'interno racchiudono tutti i riferimenti per il completamento del lavoro su di uno specifico soggetto.
Per generare delle pagine Html è necessario creare un progetto, sia per la generazione a runtime sia per la generazione manuale.

Il programma di gestione presenta l'elenco dei progetti inseriti:

Selezionando un progetto si accede alla pagina di dettaglio:

I dati previsti sono:

Nome Progetto: è il nome con cui verrà richiamato questo progetto.
Root: è il nome della directory che contiene l'istanza NetAcross e al suo interno sono contenuti gli oggetti necessari al runtime. L'installazione crea una root di default con nome NetAcross e l'utente è libero di generarne altre.
Applicazione: all'interno della root si possono creare più strutture legate ognuna ad una applicazione. I programmi all'interno della stessa applicazione possono essere raggiunte da un unico menù.
Personalizzazione: per ogni applicazione si possono definire più personalizzazioni. La ricerca delle pagine Html viene effettuata prima nelle personalizzazioni e se non esiste si passa alla directory base. Le personalizzazioni possono contenere versioni speciali per il Web, personalizzazioni legate a più aziende installate, ecc.

L'applicazione poi richiede l'elenco delle librerie che contengono oggetti DSPF. Da queste informazioni viene creata una struttura di cartelle per ospitare gli Html convertiti. Viene creata una cartella per ogni applicazione nella directory Pages e all'interno una cartella per ogni libreria digitata. La struttura viene allineata automaticamente.

Creazione Pagine HTML da DDS AS/400 [2.]

Questa funzione sulla base di un oggetto DSPF fornito, produce le pagine HTML corrispondenti per una corretta esecuzione. Tali pagine sono funzionanti in prima istanza. E' possibile fornire informazioni più specifiche per una corretta rappresentazione in due modi: inserendo delle direttive NetAcross (campo TEXT) nel sorgente DSPF oppure modificarle tramite un editor Html per renderle graficamente più gradevoli e funzionali.
Per la generazione della pagine Html viene richiesto quindi:

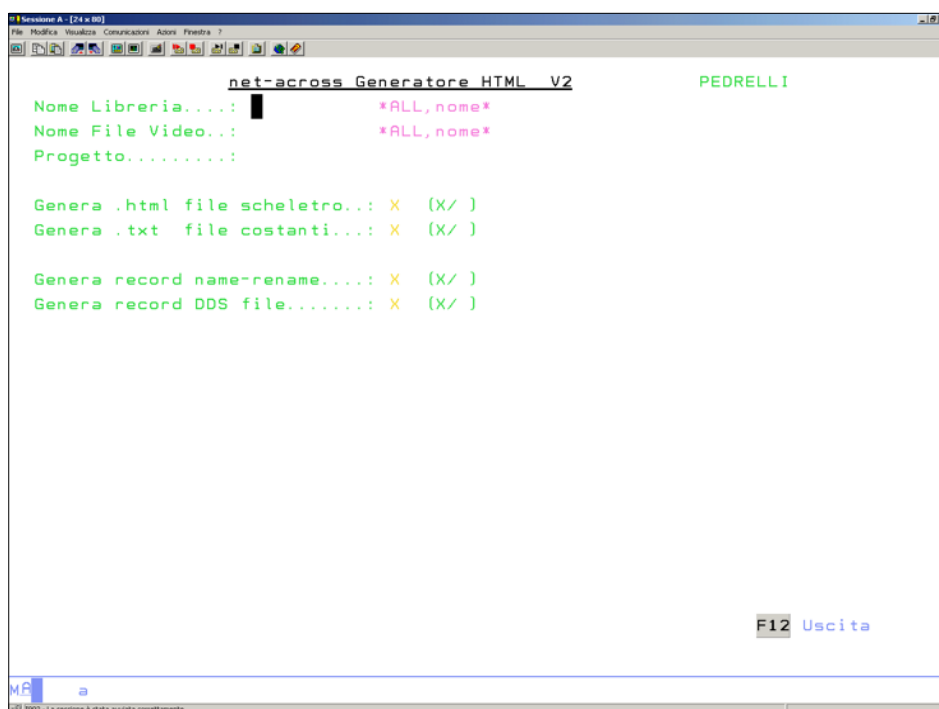
- La libreria ed il nome dell'oggetto File di tipo DSPF.
- Il nome del progetto.
- Quali oggetti creare:

  Pagine Html
  Costanti per le traduzioni
  Riferimenti a DataBase per i nomi campi
- Quali formati includere nella pagina (spesso le pagine 5250 sono composte da più formati) e quale è quello su cui viene effettuata la lettura (EXFMT).
- La conferma delle directory IFS dove creare gli oggetti.

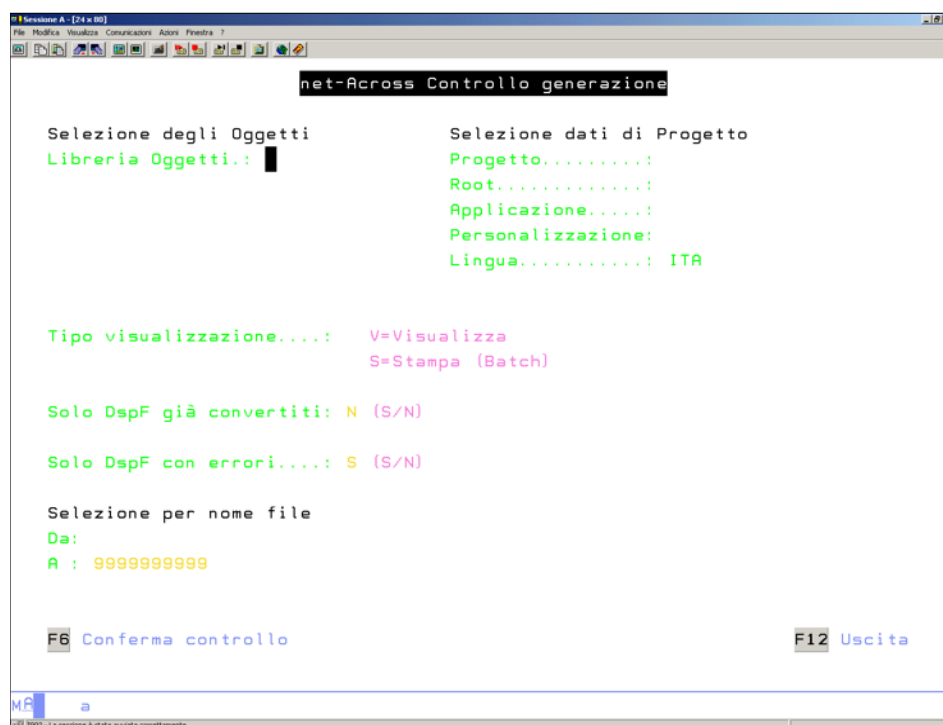
Per ogni oggetto convertito viene registrato un movimento in un file di log che consente di mantenere lo storico dei lavori. Tale log può essere consultato con un'altra opzione del menù.

Controllo Generazione [3.]

Il lavoro di generazione pagine Html a partire dai formati 5250 dell'utente, necessario al funzionamento di NetAcross, è una fase che può essere eseguita manualmente oppure sfruttando

```
                    net-across Generatore HTML  V2           PEDRELLI
 Nome Libreria....:              *ALL,nome*
 Nome File Video..:              *ALL,nome*
 Progetto.........:

 Genera .html file scheletro..: X  (X/ )
 Genera .txt  file costanti...: X  (X/ )

 Genera record name-rename....: X  (X/ )
 Genera record DDS file.......: X  (X/ )

                                                        F12 Uscita
```

la generazione "al volo" durante l'esecuzione dei programmi stessi.
Può essere necessario verificare lo stato di generazione di una libreria, per verificare lo stato di allineamento degli oggetti in seguito a delle modifiche oppure per verificare gli oggetti mai generati.
Una volta evidenziati gli oggetti non allineati è possibile eseguire sugli stessi le azioni opportune per correggere la situazione in modo anche massivo.
I dati richiesti sono i seguenti:

Libreria oggetti: il nome della libreria contenente gli oggetti di tipo *DSPF.
Progetto: il nome del progetto che indicherà dove reperire gli oggetti oggetto della generazione (html e txt costanti).
Tipo Visualizzazione: se visualizzazione sarà possibile eseguire le azioni sugli oggetti non allineati. Se stampa verrà prodotto un tabulato di sola consultazione.
Solo DspF già convertiti: mostra oppure no solo i file video per cui è già stata effettuata la conversione a pagina Html.
Solo DspF con errori: mostra oppure no solo i file video per cui c'è una segnalazione di errore.
Selezione per nome file: permette di limitare la ricerca degli oggetti *DSPF nella libreria, utile soprattutto nel caso di librerie con tanti oggetti video.

Dopo l'elaborazione dei dati se richiesta visualizzazione verrà proposta la seguente situazione:

In testata viene riportata la libreria degli oggetti *DSPF (digitata in precedenza) e la path IFS delle pagine Html (derivata dal progetto).
Del dettaglio vengono riportati gli oggetti che risultano non allineati (nome file e nome formato).

Nella primo gruppo di colonne viene riportato lo stato dei riferimenti a DataBase. Per ogni formato video viene aggiornato un gruppo di files (NADDS* nella libreria NETAF) con l'elenco dei campi, le condizioni, gli indicatori, le parole chiave, ecc.). Per ogni formato viene riportata la data e ora dell'oggetto *DSPF che lo ha generato. Se l'oggetto viene modificato viene quindi riportata la segnalazione di anomalia. Da evidenziare che essendo la data di riferimento sull'oggetto del

file video se questo viene modificato vengono marcati modificati tutti i formati, anche se realmente non lo sono stati tutti.
Nella colonna DDS le azioni previste sono:
Conversione: provvede a ricreare i dati del DataBase in riferimento al formato richiesto.
Allinea Date: se si è sicuri che il DataBase ha già i dati corretti si può forzare l'impostazione della data e ora dell'oggetto senza ricreare i dati.

Nel secondo gruppo di colonne viene riportato lo stato delle pagine Html. Per ogni pagina 5250 (eventualmente composta da più formato video) viene creata una pagina Html contenente la presentazione grafica. Per ogni pagina viene riportata la data e ora dell'oggetto *DSPF che lo ha generato. Se l'oggetto viene modificato viene quindi riportata la segnalazione di anomalia. Da evidenziare che essendo la data di riferimento sull'oggetto del file video se questo viene modificato vengono marcati modificati tutti i formati, anche se realmente non lo sono stati tutti.
Nella colonna Html le azioni previste sono:
Conversione: provvede a ricreare la pagina Html in riferimento al formato richiesto. Verrà richiesta la composizione dei formati video costituenti la pagina e il nome del formato di lettura dal programma.
Conversione Automatica Html: provvede a ricreare la pagina Html in riferimento al formato richiesto se già è a DataBase la composizione dei formati della pagina (da conversioni precedenti). In questo caso non viene richiesta la composizione dei formati e la generazione viene effettuata automaticamente. Se il formato non è quello di lettura dal programma non viene eseguito nulla.
Allinea Date: se si è sicuri che l'Html ha già i dati corretti si può forzare l'impostazione della data e ora dell'oggetto senza ricreare la pagina.

Le scelte possono essere messe in modo massivo anche su tutte le righe.

Gestione Keyword Esterne

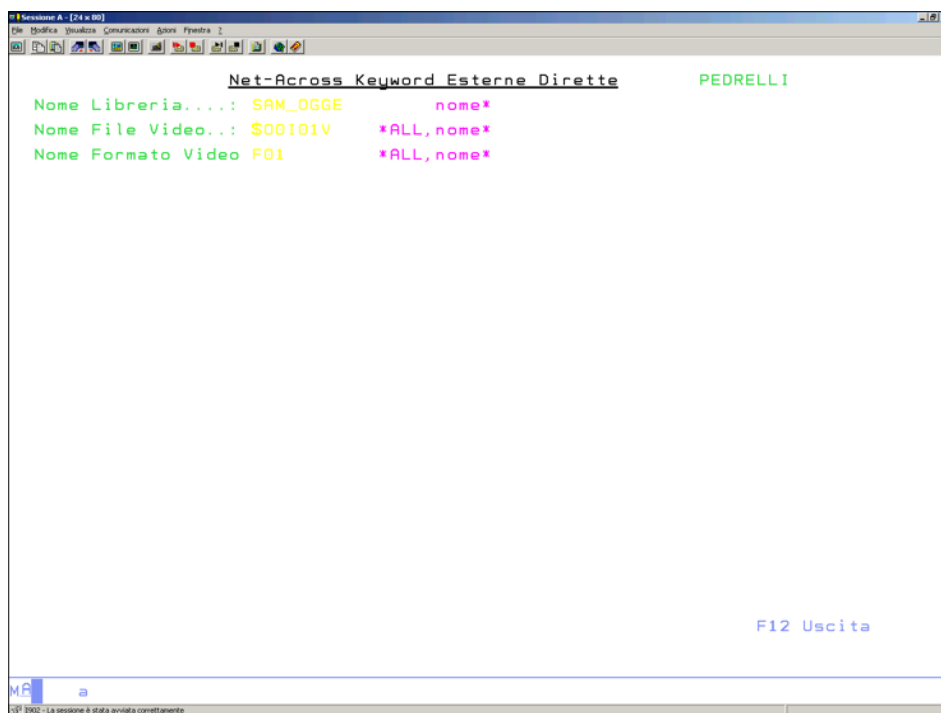
NetAcross provvede a trasformare i pannelli 5250 in pagine Html da presentare in ambiente browser. Si può migliorare il lavoro della trasformazione utilizzando delle keyword specifiche di NetAcross che danno al processo di conversione le indicazioni aggiuntive per affinarne il lavoro.
Le keyword possono essere inserite direttamente nelle DDS oppure possono essere indicate in appositi data base esterni.
Inserire le keyword nelle DDS dà il vantaggio di gestirle contestualmente alla definizione dei formati video sia utilizzando il SEU che l'SDA e quindi il programmatore può provvedere con lo stesso strumento a gestire i campi tradizionali sia a specificarne le keyword.
Inseriere le keyword nel data base esterno dà il vantaggio di poterle inserire anche se non si posseggono i sorgenti dei formati video oppure se si vuole avere la flessibilità d'uso di un file di data base anziché dei files sorgenti.
Esistono due tipi di data base esterno:
Specifico. In questo caso le keyword vengono imputate direttamente ad un campo di un formato video. La funzionalità è quindi sostanzialmente la stessa di inserire le keyword nelle DDS.
Generico. In questo caso le keyword vengono imputate in modo generico ai campi dei file video, indicando dei nomi che possono essere generici (espressioni regolari) oppure che specificano i campi di riferimento dei campi (anche qui con espressioni regolari). Ad esempio posso dire che un campo (codice articolo) verrà gestito con una funzione estesa che permetterà di vedere la scheda tecnica e valutare la disponibilità. Con il data base generico non devo indicare il nome specifico che il campo ha nello specifico formato video (come si farebbe con le keyword DDS oppure le keyword con data base specifico) ma posso indicare *CDAR (tutti i campi che finiscono con CDAR) oppure tutti i campi il cui campo di riferimento si chiama CDAR o *CDAR.

Le keyword una volta inserite sono considerate pendenti e verranno applicate la prossima volta che verrà richiesta la trasformazione in pagina Html (è modificato il formato video oppure viene richiesto dal Controllo Generazione. E' possibile forzare la trasformazione delle keyword pendenti coinvolgendo anche tutti i formati video già convertiti.

Gestione Keyword Esterne (Specifiche) [11.]

Questa funzione permette di inserire le keyword riferite a campi specifici. Il primo pannello presenta quindi i campi per specificare il nome del formato video. Nella selezione viene richiesto la libreria, il nome del file video ed il nome del formato video. In tutti i campi è possibile indicare un nome generico per facilitare la selezione.

Una volta specificato il formato video vengono richieste le keyword:

Il primo campo è il formato record ed anche su questo è possibile indicare le keyword. In colore diverso vengono rappresentati i campi che hanno delle variazioni che devono ancora essere applicate.
Per ogni campo è possibile inserire una o più keyword.

Gestione Keyword Esterne (Generiche) [21.]

Questa funzione permette di inserire le keyword riferite genericamente ai campi dei formati video. Il primo pannello presenta quindi i campi per specificare il nome del progetto ed i dati della keyword:

```
                Net-Across Keyword Esterne per Riferimento   PEDRELLI
 Nome Progetto....: COMMA
 Nome Istanza.....: NETACROSS
 Nome Applicazione: COMMA
 Nome Personalizz.:

 Tipo Riferimento.:
 valore...........:

 priorità.........:

 Tipo:  1 - campo singolo diretto
        2 - campo singolo con espressione regolare
        3 - riferim.dizionario LIBRERIA(10)+FILE(10)+RECORD(10)+CAMPO(10)
        4 - riferim.dizionario per CAMPO con espressione regolare

                                                              F12 Uscita
```

Nome progetto è il nome del progetto a cui fa riferimento la keyword. Vedere Gestione Progetti in questo manuale.
Tipo riferimento indica con quale modalità è indicato il campo che segue (valore). Può assumere i valori:
1 - campo singolo diretto. In questo caso il valore dovrà contenere il nome completo del campo a cui si fa riferimento.
2 - campo singolo con espressione regolare. In questo caso il valore dovrà contenere una espressione regolare che identifica il campo/i campi a cui si fa riferimento.
3 - riferim.dizionario LIBRERIA(10)+FILE(10)+RECORD(10)+CAMPO(10). In questo caso il valore dovrà contenere il nome completo del campo di dizionario a cui il campo fa riferimento.
4 - riferim.dizionario per CAMPO con espressione regolare. In questo caso il valore dovrà contenere una espressione regolare che indica il campo/i campi di dizionario a cui il campo fa riferimento.
Priorità indica nel caso di coincidenza di più valori quale deve avere la precedenza.

Se non si indicano valori si ottiene l'elenco delle keyword già inserite:

In colore diverso vengono rappresentati i campi che hanno delle variazioni che devono ancora essere applicate.

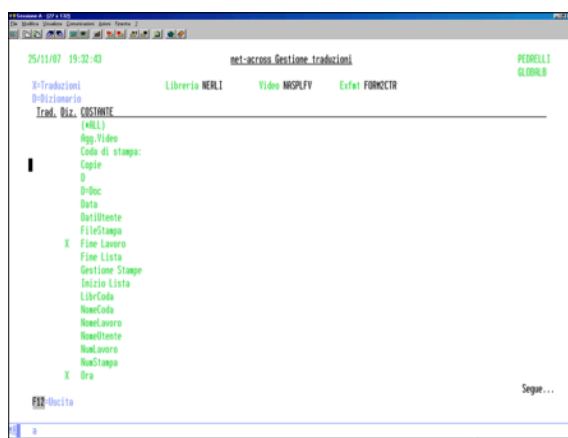
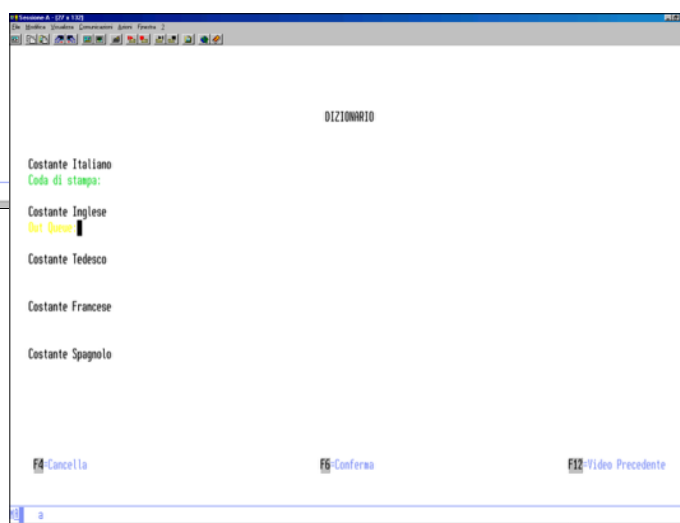
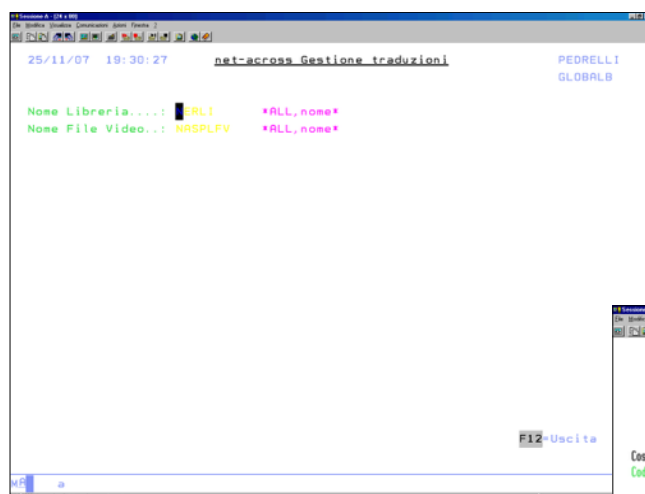
Conversione Massiva Keyword Esterne [22.]

Questa funzione permette di applicare le keyword esterne agli oggetti NetAcross. Le keyword quando vengono inserite sono attive solo nei confronti delle generazioni Html future. Questa opzione provvede ad applicare le nuove regole anche alle generazioni già effettuate.
La pagina seguente indica i dati da inserire:

Gestione traduzioni [30.]

L'apertura al mondo Internet richiede spesso che le applicazioni debbano funzionare in un ambiente multilingue.
Se l'applicazione già lo è questo è trasparente a NetAcross che quindi funzionerà regolarmente.
Se l'applicazione non lo è NetAcross fornisce con questa opzione il supporto per le traduzioni.
Tutte le costanti a video possono essere facilmente tradotte con questa funzione. Le traduzioni possono essere inserite come dizionario, ed in questo caso automaticamente traducono tutte le frasi identiche a quella tradotta; oppure possono essere inserite come traduzione specifica ed in questo caso valgono solo per il contesto in cui siamo.
Le traduzioni si riferiscono alla parte costante specificata nei formati video. La parte emessa a contenuto variabile (messaggi, campi di output, ecc.) rimane a responsabilità dell'applicazione utente.
Digitare nella pagina che segue il nome del file video (oggetto) e la relativa libreria. Sono

disponibili le ricerche per nome totale o generico.

Selezionato il file video viene proposto l'elenco dei campi che lo compongono. Da qui sarà possibile selezionare X=traduzione per effettuare una traduzione che vale solo per questo formato video oppure D=Dizionario per effettuare una traduzione che vale per tutte frasi identiche a questa in tutti i formati video.

Selezionato il campo sarà possibile inserire le traduzioni nelle varie lingue.

Archivi memorizzazione DSPF NetAcross

Ogni file video gestito da NetAcross determina la memorizzazione dei suoi dati in una struttura di archivi all'interno della libreria NETAF. Questi archivi hanno il nome che inizia con NADDS e al loro interno vengono memorizzate le varie informazioni che contiene un file video (nomi dei campi, indicatori, keywords, ecc.), le relazioni fra record e le traduzioni in lingua.
All'interno dei files NADDS* ce n'è una parte dedicata ai singoli formati video con i quali c'è un rapporto diretto.
NADDS01F contiene una riga per ogni campo del file video e NADDS02F contiene una o più righe per ogni keyword DDS o NetAcross presenti sul campo.
NADDS04F contiene l'elenco dei campi relativi alle singole composizioni video (una pagina video spesso è composta da formati diversi e NetAcross la gestisce come pagina unica) e NADDS06F contiene l'elenco delle possibili composizioni.
NADDS05F contiene una riga per ogni libreria e permette di gestire librerie che a livello di sistema operativo si chiamano con un nome e a livello di NADDS* sono state convertite con un altro nome. Questa funzione è di uso prevalente per le software house.
NADDS07F contiene una riga per ogni campo costante e può quindi contenere le relative traduzioni in lingua.

Struttura Directory istanza NetAcross

Ogni istanza di Http Server deve avere una cartella di riferimento nella root del IFS dell'iSeries con lo stesso nome dell'istanza.
All'interno di tale cartella deve esistere una struttura di cartelle preordinata e indipendente dall'applicazione ivi caricata.
La struttura delle cartelle viene fornita nel CD di installazione ed in alcune parti è possibile personalizzare le informazioni (logo aziendali, CSS, script personalizzati, ecc.).
Andiamo ad analizzare le cartelle:

Cartelle indipendenti dall'applicazione

/<istanza>/CONFIG contiene i file di configurazione. Contiene il file di configurazione generale **config.xml** ed il template (**template.html**) e le primitive (**primitive.xml**) per la generazione delle pagine Html.

/<istanza>/STYLE contiene i fogli di stile (CSS) che standardizzano gli elementi grafici delle pagine Html per uniformare le pagine Web. Sono disponibili più stili in funzione delle diverse risoluzioni video.

/<istanza>/IMAGES contiene gli elementi grafici usati dal prodotto NetAcross.

/<istanza>/SCRIPTS contiene i codici javascript usati dall'applicazione. Se si ha necessità di aggiungere dei javascript personalizzati nelle pagine, si consiglia di utilizzare un file javascript

separato, inserirlo nella directory e richiamarlo con la keyword *JSC.

/<istanza>/SERVICE contiene pagine di servizio utilizzate dal prodotto NetAcross.

/<istanza>/TOOLBAR contiene immagini, css, ecc. utilizzate dal prodotto NetAcross per la creazione della Toolbar.

/<istanza>/PF2XLS contiene il codice necessario alla generazione di fogli di calcolo da subfile o da API di programmazione NetAcross.

/<istanza>/SP2DOC contiene il codice necessario alla generazione di documenti da spool o da API di programmazione NetAcross.

/<istanza>/SP2PDF contiene il codice necessario alla generazione di documenti PDF da spool o da API di programmazione NetAcross.

/<istanza>/AS2MAIL contiene il codice necessario alla generazione delle e-Mail da programma.

/<istanza>/SAMPLES contiene degli esempi di configurazione riutilizzabili dall'utente. In sede di installazione viene assunto automaticamente il file di configurazione NetAcross ed il file di configurazione dell'Http Server.

/<istanza>/PERS deve contenere tutti gli oggetti forniti da NetAcross che sono stati modificati dall'utente, rispettando le eventuali sottodirectory. Questo garantisce che in fase di aggiornamento del prodotto vengano rispettate le personalizzazioni e che queste non vengano ricoperte dalla nuova versione.

/<istanza>/ORIG viene riempita automaticamente dall'installazione NetAcross ed a seguito di questa conterrà tutti gli oggetti originali che sono stati rimpiazzati da oggetti personalizzati (cartella PERS). Questo permette di disporre anche degli oggetti originali e poterli confrontare con i personalizzati in caso di malfunzionamenti.

/<istanza>/PAGES è usata per contenere la strutture delle pagine html dell'applicazione e dei relativi files correlati. Tale struttura contiene un albero per ogni applicazione servita, e verrà dettagliata nel prossimo capitolo.

/<istanza>/XLS è usata per contenere la strutture dei template xls dell'applicazione per la generazione automatica da subfile. Tale struttura contiene un albero per ogni applicazione servita, e segue le regole della directory PAGES.

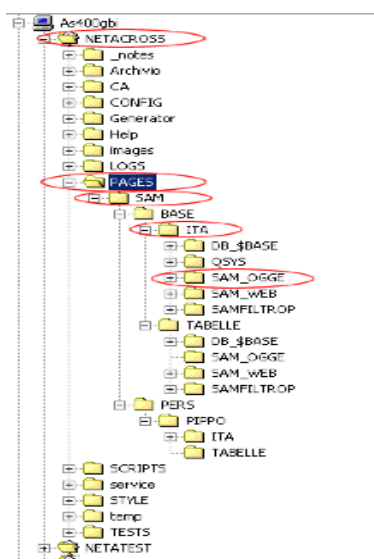
/<istanza>/LOGS contiene le log dell'Http Server, divise fra log degli accessi, log degli errori e log degli errori CGI. La sua cancellazione periodica è specificata nel file di configurazione dell'Http Server.

/<istanza>/GENERATOR è una directory di servizio del generatore e può essere usata per generare HTML di prova.

/<istanza>/TEMP è una directory di servizio e serve se abilitata la funzione GZIP per effettuare le compressione delle pagine Html.

Cartelle dipendenti dall'applicazione

Ogni applicativo dominio di NetAcross deve fare riferimento ad una struttura directory memorizzate sull'IFS dell'iSeries che conterranno gli HTML da utilizzare per presentare le videate As/400 (struttura PAGES).
Nell'esempio seguente si può notare che la root NETACROSS (stesso nome dell'istanza Http Server) contiene una cartella PAGES (nome fisso) che a sua volta ha al primo livello la cartella SAM (esempio di applicazione, nome collegato nell'utente della Validation List).
All'interno c'è la cartella ITA (i files base in italiano) e poi le cartelle che corrispondono alle librerie AS/400 in cui ci sono i DSPF dell'applicazione.

Riassumendo (caso utente Internet): un utente con nome samweb connesso tramite il suo browser verrà riconosciuto da un'istanza HTTP (NETACROSS) e associato al profilo utente di Logon stabilito dalla configurazione (PROVAWEB) e da questo riceverà lista librerie e gli altri attributi. Dal primo utente (SAMWEB) riceverà l'informazione circa l'applicativo (SAM), la personalizzazione (nessuna) e la lingua (italiano). Le pagine Html per tale utente saranno ricercati nella cartella /NETACROSS/PAGES/SAM/BASE/ITA/*.*

l'uso dell'area PERS può permettere di modificare gli HTML senza perdere gli originali e inserire in essi alcuni elementi non standard, per differenziare gli utenti, per esempio, o per fornire particolari funzionalità non attivate sullo standard.

All'interno della cartella PAGES deve esistere quindi una cartella per ogni applicativo gestito.
All'interno dell'applicativo deve esistere la cartella BASE; facoltativamente la cartella PERS, che ospiterà le eventuali personalizzazioni.
All'interno di quest'ultime deve esistere la cartella ITA. Possono esistere delle cartelle di traduzioni ING, TED, FRA oppure SPA.
All'interno delle cartelle ITA deve esistere una cartella per ogni libreria dell'applicazione, al cui interno verranno inserite le maschere Html convertite.

Interfaccia per il programmatore

API

API per identificazione utenti NetAcross (Pgm NETAINFOP)

All'interno delle applicazioni utente può essere necessario reperire i dati dell'utente richiedente per determinare le funzioni disponibili. Ad esempio se un utente si collega via Internet e l'accesso è con un identificativo che riconduce ad un cliente o un agente, dovranno essere ridotte le funzionalità nell'ambito di sè stesso nel caso del cliente e di sè stesso ed i propri clienti nel caso dell'agente.
Allo stesso modo si potrà limitare l'accesso alla modifica di dati come prezzi, sconti, ecc. Oppure a tasti di comando, ricerche.
Le informazioni relative agli utenti collegati allo specifico programma possono essere recuperate all'interno delle applicazioni tramite un'interfaccia standardizzata che è resa disponibile da NetAcross.
Tali informazioni dovranno quindi essere utilizzate per inibire o meno determinate funzionalità per tipo utente o per utente specifico.
L'interfaccia specifica è il programma **NETAP/NETAINFOP**.
Una volta chiamato il programma saranno disponibili le informazioni sull'utente connesso al web server per la sessione corrente.
L'interfaccia prevede i seguenti parametri:

VTNAME 10 caratteri alfabetico.
WWESITO 1 carattere alfabetico.
VTUSER 30 caratteri alfabetico.
VTTYPE 3 caratteri alfabetico.
VTCODE 6 caratteri alfabetico.
VTDESC 30 caratteri alfabetico.
VTENV 3 caratteri alfabetico.
VTAPPL 10 caratteri alfabetico.
VTPERS 10 caratteri alfabetico.
VTLANG 3 caratteri alfabetico.
VTSINF 10 caratteri alfabetico.

I parametri hanno questo significato:

- VTNAME : nome del terminale video sul quale cui è in esecuzione il programma, quindi il nome del job interattivo per la sessione.
- WWESITO : flag che identifica se la sessione è in esecuzione via sessione Web oppure è tramite un terminale normale (ClientAccess o Terminale), ritorna spazio se sessione Web altrimenti ritorna 'E' nel caso di una sessione da terminale 5250. Nel caso di sessione 5250 tutti i parametri seguenti saranno restituiti a spazi.
- VTUSER : nome dell'utente come inserito nel browser al momento della connessione, corrisponde ad Utente Web nella configurazione.

- VTTYPE : tipo di utente (AGE=Agente; CLI=Cliente; FOR=Fornitore; ALL=nessuna limitazione).
- VTCODE : codice interno al database per l'utente (se AGE, CLI o FOR) .
- VTDESC : descrizione dell'utente.
- VTENV : indica se la sessione è interattiva (INT) o batch (BCH).
- VTAPPL : applicazione Web specificata nella configurazione.
- VTPERS : eventuale personalizzazione.
- VTLANG : lingua impostata per l'utente (ITA/ING/TED/FRA/SPA ecc..).
- VTSINF : sistema informativo impostato per l'utente.

API per gestione parametri richiamo da portale (pgm NETAPARMP)

Il richiamo di NetAcross da portale prevede il passaggio di parametri sia in entrata che in uscita. Questa API permette di gestirli.

L'interfaccia specifica è il programma **NETAP/NETAPARMP**
L'interfaccia prevede i seguenti parametri:

VTNAME 10charatteri alfabetico
WWESITO 1 carattere alfabetico
FLGAGG 1 carattere alfabetico
INPRM1 528 caratteri alfabetico

I parametri hanno questo significato:

- VTNAME : nome del terminale video sul quale cui è in esecuzione il programma, quindi il nome del job interattivo per la sessione.
- WWESITO : flag che identifica se la sessione è in esecuzione via sessione Web oppure è tramite un terminale normale (ClientAccess o Terminale), ritorna spazio se sessione Web altrimenti ritorna 'E' nel caso di una sessione da terminale 5250. Nel caso di sessione 5250 tutti i parametri seguenti saranno restituiti a spazi.
- FLGAGG : se vale 1 il valore INPRM1 viene usato come aggiornamento, altrimenti viene usato per leggere il valore.
- INPRM1 : parametro.

API per attribuzione Utente Validation List a stampe pdf

La gestione PDF quando è utilizzata con utenti di Validation List per riuscire a collegare i singoli PDF agli utenti specifici necessita di informazioni ulteriori per effettuare il collegamento. Il file PDF infatti è collegato all'utente di sistema, mentre una Validation List condivide per più utenti lo stesso utente (User Profile).
Per fare questo occorre modificare i programma che creano file di spool destinati alla creazione di PDF modificando il nome spool (parametro SPLFNAME del comando OVRPRTF) con il valore ritornato dalla presente API.

Esistono due API da richiamare:
NETAINFSP quando si vogliono generare PDF permanenti, da usare o con la Gestione PDF oppure con la Gestione Documenti.
NETAINFSAP quando si vogliono generare PDF "al volo" ed assicurarne quindi la cancellazione, come descritto in Lancio automatico di documenti – PDF in questo manuale.
Il programma va richiamato fornendogli due parametri:
Esito, di 1 carattere alfabetico.
N.Sessione, di 10 caratteri alfabetico.
Entrambi i parametri vengono inizializzati dal programma, per cui non importa con quali valori vengono forniti.
I valori di ritorno valgono come segue:
Esito = E. Errore, il lavoro non è stato riconosciuto come lavoro NetAcross. N.Sessione vale zero.
Esito = S. Il lavoro è riconosciuto come lavoro NetAcross, però l'utente non è un utente di Validation List ma di sistema, per cui non occorre la modifica del nome spool. N.Sessione vale zero.
Esito = spazio. Il lavoro è riconosciuto come lavoro NetAcross, l'utente è un utente di Validation List. Il N.Sessione è il valore da inserire nel parametro SPLFNAME per sostituire il nome Spool da creare. Il nome è già comprensivo del prefisso NA o NR richiesto.

API per lettura Utenti Validation List

Gli utenti per poter accedere alle applicazioni via browser devono essere iscritti, e le iscrizioni sono memorizzate in Validation List. Poiché leggere le Validation List comporta l'utilizzo di API del sistema operativo abbastanza complesse questa API fornisce un'interfaccia per poter leggere più facilmente le informazioni contenute.
Le informazioni si ottiengono chiamando il programma NetAcross **READVALL** con i seguenti parametri:

ValList è il nome della Validation List, che viene ricercata nella libreria NETAF. Parametro 1: 10 caratteri alfabetico.
User è il nome utente da ricercare. Deve essere inserito solo se TypeRead = U. Parametro 2: 30 caratteri alfabetico.
TypeRead è il tipo di ricerca da effettuare. U=per utente. Inserire l'utente nel parametro User. F=First. Il primo elemento della lista. N=Next. Il successivo elemento della lista. Parametro 3: 1 carattere alfabetico.
UserRtn è il nome utente letto. Parametro 4: 30 caratteri alfabetico.
Dati sono i dati relativi all'utente letto. Parametro 5: 256 caratteri alfabetico. Segue contenuto del campo dati:

```
WWType   3A  Tipo utente
WWCode   6A  Codice gestionale
WWDesc  30A  Descrizione
WWLgUt  10A  Utente di Logon
WWLgMn  10A  Menù di Logon
WWLgPg  10A  Programma di Logon
WWEnv    3A  Ambiente INT/BCH
```

WWAppl 10A Applicazione
WWPers 10A Personalizzazione
WWLang 3A Lingua ITA/ING/TED/FRA/SPA
WWGZip 1A Uso compressione Y/N/A
WWAuth 2A Livello autorità
WWUtOb 20A Utilità oggetto
WWCSS 30A Path CSS utente
WWTerm 10A Nome terminale 5250

Esito è l'esito della chiamata. E=Errore. Spazio=OK. Parametro 6: 1 carattere alfabetico.
ErrNo è il codice di errore nel caso di Esito=E. Il codice di errore fa riferimento ai messaggi CPExxxx dove xxxx è il codice di errore. Tali messaggi sino nel file messaggi QCPFMSG nella libreria QSys. Parametro 7: 4 numerico tipo S 0 decimali.

API per Cambio Password

Gli utenti NetAcross possono essere User Profile di sistema oppure utenti inseriti in Validation List. Se è necessario cambiare la password da programma lo si può fare richiamando questa API che provvede a tenere le password sincronizzate fra utente di sistema e NetAcross.
Il cambio password si ottiene chiamando il programma NetAcross **CGICHGPWD** con i seguenti parametri:

OldPass è la vecchia password, che viene controllata prima di applicare la nuova. Se il profilo è utente di sistema la password deve essere massimo 10 caratteri. Se il profilo è da Validation List la password può essere lunga 30 caratteri ed è sensibile al maiuscolo/minuscolo. Parametro 1: 30 caratteri alfabetico.
NewPass è la nuova password. Se il profilo è utente di sistema la password deve essere massimo 10 caratteri. Se il profilo è da Validation List la password può essere lunga 30 caratteri ed è sensibile al maiuscolo/minuscolo. Parametro 2: 30 caratteri alfabetico.
RmtUser è il nome dell'utente da modificare. Parametro 3: 30 caratteri alfabetico.
VldList è il nome della Validation List. Viene sempre cercata nella libreria NETAF. Parametro 4: 10 caratteri alfabetico.
SystemUser indica se l'utente è di sistema o di Validation List. Se è di sistema viene cambiato automaticamente utente di sistema e Validation List. Indicare 1 per sistema e 2 per Validation List. Parametro 5: 1 carattere alfabetico.
Esito è il risultato della API. Se tutti spazi l'operazione è andata a buon fine. In caso contrario viene riportato il codice dell'errore. Parametro 6: 10 caratteri alfabetico.

API per identificazione azioni interattive/batch (Pgm NARTVTPJOP)

Questa API è utile solamente quando si utilizza la modalità batch.
In modalità batch per definizione tutte le azioni eseguite da un menù vengono sottomesse come lavoro batch. Può essere però importante all'interno del programma stabilire se l'azione viene

eseguita come un vecchio lavoro interattivo (colloquio con l'utente) oppure se come un vero lavoro batch (non c'è colloquio con l'utente).
A seconda dell'esito della chiamata a questa API il programma può decidere se aprire o no un formato video ed eseguire delle transazioni interattive. L'API deve quindi essere richiamata all'interno del lavoro di cui si vuole sapere l'informazione.
L'informazione si ottiene chiamando il programma NetAcross **NARTVTPJOP** con i seguenti parametri:

Esito è il risultato della chiamata. Parametro 1: 1 carattere alfabetico. E=Errore (non è un lavoro NetAcross) Spazio=Ok.
Tipo è il tipo di lavoro. Parametro 2: 3 caratteri alfabetico. INT=Interattivo BCH=Batch Spazi=Errore.

API per ritorno libreria simulazione QTEMP (Pgm NAQTEMPP)

Questa API è utile solamente quando si utilizza la modalità batch.
In modalità batch quando si esegue una azione da menù il lavoro viene eseguito e termina con il ritorno a menù. Per compatibilità verso il funzionamento 5250 i dati relativi alla QTEMP possono essere conservati automaticamente in una libreria di nome NETAW + il numero della sessione. Questa libreria può essere utilizzata dal programmatore anche per appoggiare dei propri dati che possono essere conservati appunto attraverso i richiami di voci di menù.
Questa API ritorna il nome della libreria temporanea NetAcross.
L'informazione si ottiene chiamando il programma NetAcross **NAQTEMPP** con i seguenti parametri:

Nome Libreria è il risultato della chiamata. Parametro 1: 10 caratteri alfabetico.

API per visualizzazione Spool

Questa API è utile solamente quando si utilizza la modalità batch.
In modalità batch non è disponibile il comando WrkSplf. Avendo a disposizione il nome dello spool questa API fornisce l'interfaccia di visualizzazione di un singolo spool.
La visualizzazione si ottiene chiamando il programma NetAcross **NAVISSPLP** con i seguenti parametri:

Nome Lavoro è il nome del lavoro che ha generato lo spool. Parametro 1 10 caratteri alfabetico.
Nome Utente è il nome dell'utente che ha generato lo spool. Parametro 2 10 caratteri alfabetico.
Numero Lavoro è il numero del lavoro che ha generato lo spool. Parametro 3 6 caratteri numerico.
Nome Spool è il nome del file di stampa che ha generato lo spool. Parametro 4 10 caratteri alfabetico.
Numero Spool è il numero dello spool. Parametro 5 4 caratteri numerico.
Ampiezza Pagina è l'ampiezza in colonne massima dello spool. Parametro 6 3 caratteri numerico.
Codice di Ritorno è l'esito dell'operazione. Parametro 7 intero con segno.

0 OK.
-1 Caratteri errati in struttura File di Spool.
-2 Errore buffer il lettura File di Spool.
-3 Errore in apertura File di Spool.

API per visualizzazione Messaggi

Questa API è utile solamente quando si utilizza la modalità batch.
In modalità batch non è disponibile la visualizzazione messaggi legati ad un file di stampa.
Avendo a disposizione il nome della coda di stampa o della stampante questa API fornisce l'interfaccia di visualizzazione dei messaggi a loro legati.
La visualizzazione si ottiene chiamando il programma NetAcross **NAMSGSPLP** con i seguenti parametri:

Nome Coda è il nome della coda di output di cui visualizzare i messaggi. Parametro 1 10 caratteri alfabetico.
Nome Device è il nome del device di cui visualizzare i messaggi. Parametro 2 10 caratteri alfabetico.
Codice di Ritorno è l'esito dell'operazione. Parametro 3 10 caratteri numerico.
0 OK.

API per Lista Directory (Pgm NALSTDIR)

Scorrere il contenuto di una directory nel IFS è una operazione che richiede l'utilizzo di API di sistema e quindi è una operazione abbastanza complessa.
Per semplificare il lavoro di chi interfaccia NetAcross viene fornita questa API che permette di scorrere il contenuto di qualsiasi directory IFS e verificare il contenuto anche delle directory nidificate. Ogni chiamata alla API ritorna un file per volta fino alla fine dell'elenco.
Questa operazione è utile quando si hanno ad esempio delle immagini legate ad un articolo ma non si conosce esattamente il nome di ognuna perchè sono più foto dello stesso articolo legate magari a più colori o più angolazioni. La lista fornisce quindi l'elenco delle immagini che possono essere elencate in un subfile con a lato l'immagine corrispondente.
La lista si ottiene chiamando il programma NetAcross **NALSTDIR** con i seguenti parametri:

Path è il percorso in cui effettuare la ricerca. Deve puntare ad una directory. Parametro 1 [256A]
PathL è la lunghezza in caratteri del parametro Path. Parametro 2 [10I 0]
Nome viene ritornato dalla API e contiene il nome del primo file trovato. Può essere anche una directory. Parametro 3 [256A]
NomeL è la lunghezza in caratteri del parametro Nome. Parametro 4 [10I 0]
Esito è l'esito di ogni singola richiesta. Alla prima chiamata riempirlo con 0. Parametro 5 [10I 0]
0 da settare alla prima chiamata.
1 il nome restituito è un file.
2 il nome restituito è una directory.

3 il nome restituito non è né un file né una directory.
9 fine della directory.
Errore è l'esito della chiamata. Parametro 6 [4A]
0 = OK
Altro= codice di errore. Per decodificare DSPMSGD CPEXXXX.
Ambito è l'area in cui è stato riscontrato l'errore per codici diversi da 0. Parametro 7 [64A]

API per Controllo Esistenza ed Accessibilità file IFS (Pgm NAACCESSP)

Controllare esistenza e permessi di accesso di un file nel IFS è una operazione che richiede l'utilizzo di API di sistema e quindi è una operazione abbastanza complessa.
Per semplificare il lavoro di chi interfaccia NetAcross viene fornita questa API che permette di ricevere informazioni su esistenza e permessi di accesso di qualsiasi file IFS.
La verifica si ottiene chiamando il programma NetAcross **NETAP/NAACCESSP** con i seguenti parametri:

Path è il percorso in cui effettuare la ricerca (senza percorso alias se inserito nel parametro 2). Deve puntare ad un file. Parametro 1 256 caratteri alfabetico.
Alias è il nome dell'Alias (vedere Gestione Alias/Directory in questo manuale). Parametro 2 32 caratteri alfabetico.
F viene ritornato dalla API e contiene 1 se il file esiste. Se altri valori il file non esiste. Parametro 3 1 caratteri alfabetico.
X viene ritornato dalla API e contiene 1 se il file ha permessi di esecuzione. Se altri valori il file non è eseguibile. Parametro 4 1 caratteri alfabetico.
R viene ritornato dalla API e contiene 1 se il file ha permessi di lettura. Se altri valori il file non non è leggibile. Parametro 5 1 caratteri alfabetico.
W viene ritornato dalla API e contiene 1 se il file ha permessi di scrittura. Se altri valori il file non è scrivibile. Parametro 6 1 caratteri alfabetico.

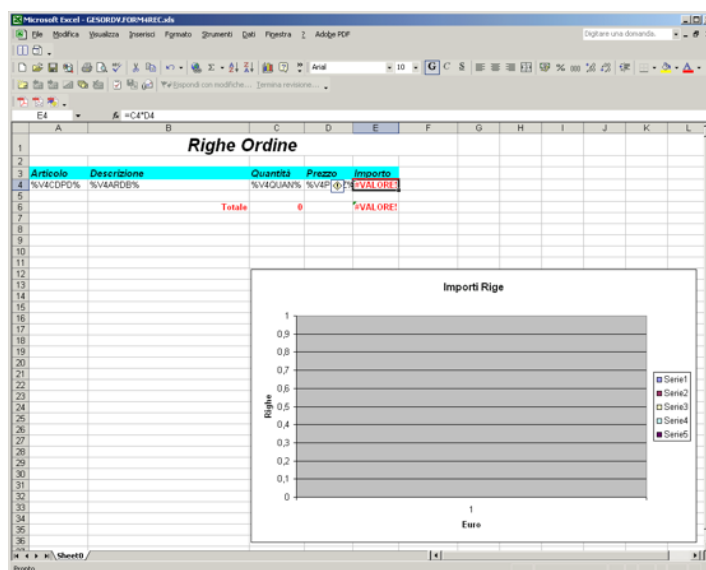
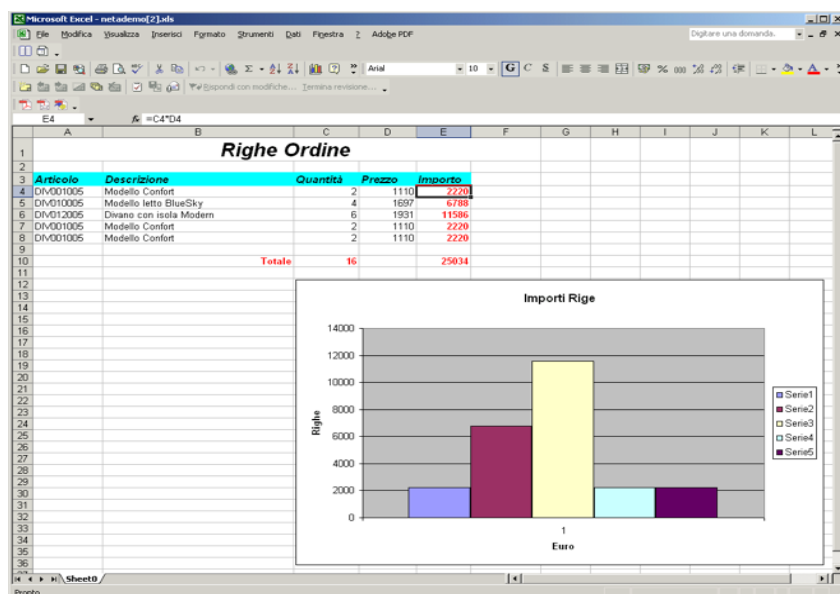
API Creazione Fogli di Calcolo da file AS400 (Cmd PF2XLS)

Molti utenti sono abituati a trattare i dati in formato foglio di calcolo (ad esempio Excel).
Un'apposito tool di NetAcross permette la creazione di questo formato a partire da un file di DataBase AS400.
Con NetAcross, senza bisogno di programmazione, è disponibile la possibilità di intregrare i dati di tutti i subfile in foglio di calcolo mediante la keyword *XLS (vedere Direttive HTML nelle DDS in questo manuale). Se si vogliono eseguire funzioni più personalizzate o più automatizzate è possibile eseguirle tramite la presente API come spiegato di seguito.
Esistono diversi tool che permettono il semplice travaso di dati. Il tool NetAcross permette di partire da un foglio di calcolo già preformattato e di andarne a riempire le caselle preservandone l'aspetto e le componenti attive (formule) riparametrizzandole in funzione dei dati inseriti.
Il meccanismo di creazione prevede che venga fornito un foglio di calcolo di base (template) al cui interno è stato riservato il posto per i dati del database. Questi posti vengono riservati scrivendo nella cella il nome del campo circondato dal carattere % (per fare un esempio il campo CODCLI deve essere indicato come %CODCLI%). Possono essere indicati i campi del subfile come anche i

campi della pagina esterni al subfile (ovviamente posizionandoli ognuno con la propria logica).
Per problemi di sintassi Html ed in entrambi i casi se nei nomi campi è stato usato il carattere £ (Lira) questo deve essere sostituito col carattere L; se è stato usato il carattere § (Paragrafo) questo deve essere sostituito col carattere S.
A seguito dell'esecuzione del tool viene generato un foglio di calcolo con i dati risultanti dall'accesso ai dati richiesti. Tale foglio può essere facilmente proiettato all'utente all'interno del browser Web come una pagina separata che ha tutta la funzionalità di Excel. Ovviamente sul client deve essere installato o l'Excel di Microsoft oppure un programma che abbia la capacità di rappresentarne il contenuto (ad esempio il prodotto gratuito OpenOffice).
E' possibile inserire formule che fanno riferimento ai dati del database, sia in senso orizzontale (formula interna alle righe database che fa riferimento a celle nella stessa riga) sia in senso verticale (formula esterna alle righe database che fa riferimento a celle della stessa colonna). Tali formule verranno ricalcolate automaticamente in funzione del numero delle righe inserite realmente dal Database.

Stringa SQL

E' possibile indicare in modo specifico la stringa SQL da richiamare per il reperimento dei dati. Tale stringa potrà contenere le variabili di sostituzione &L per la libreria e &F per il nome del file da richiamare, passabili tramite parametri al tool. Tale stringa SQL deve essere inserita in una qualsiasi cella del foglio preceduta e seguita dai caratteri %%. Tale cella nel foglio risultante verrà svuotata. Se la stringa SQL non viene specificata, ne viene eseguita una di default, contenente tutti i campi e con l'ordinamento fisico del file.
La stringa SQL può essere anche essa stessa una formula di concatenazione con risultato una stringa formattata con i doppi percentuale, in modo da rendere parametrico l'SQL in base a parametri passati, per esempio nei dati di testata.
Ogni foglio inserito nel file Excel verrà eseguito separatamente, consentendo di effettuare più accessi SQL su archivi separati. La stringa SQL permette qualsiasi operazione di Join, totalizzazione, ordinamento, ecc. come permesso dalla sintassi SQL AS400.

Esempio:
Nel template che segue è stata inserita la righe 1-3 di intestazione, predisponendo la larghezza delle colonne (esempio il campo testo largo) ed il colore (il testo rosso).
La riga 4 prevede la sostituzione con campi di database, ogni nome circondato dal carattere %. L'ultima cella a destra è una formula e prevede la moltiplicazione dei valori precedenti. Anche le celle di sostituzione del Database possono avere i propri stili e colori
Le righe 5 e 6 sono le righe di coda, e in particolare è stata inserita una formula per totalizzare la colonna del numero dei record (=SOMMA(F2:F2)).

NEW! da gennaio 2013 è possibile generare anche documenti in formato .xlsx con la stessa keyword, indicando nel file destinazione appunto un nome con estensione .xlsx. Se si fa uso del templare è obbligatoria che il tipo file del templare sia coerente con il file in uscita, e **non verrà eseguita** nessuna conversione tra i due formati.
Dopo aver eseguito il tool il risultato sarà il seguente (ovviamente a seconda del contenuto del file):

Per creare fogli di calcolo il comando è:

NETAP/PF2XLS

con i seguenti parametri:

Root E' il nome della directory di installazione NetAcross. Normalmente è '/NetAcross'. Deve essere inserita la barra iniziale e non deve essere inserita quella finale.
PathTmp E' il nome del foglio di calcolo da utilizzare come template. Il file deve risiedere in una directory del IFS AS400. Se viene fornita una stringa vuota o a tutti spazi viene utilizzato un template standard che prevede una testata generica e come intestazione delle colonne il nome del campo.
Il valore special ***CREATEMPLATE** permette di generare un template vuoto partendo da un file di database.
PathFin E' il nome del foglio di calcolo che verrà creato come risultato finale. Il file risiederà in una directory del IFS AS400. Attenzione: se il file già esiste verrà sostituito senza segnalazione.
NomeSistema E' il nome del sistema su cui sarà eseguita la stringa SQL per la lettura dei files. Può essere l'AS400 locale (specificare *LOCAL) oppure un sistema (anche non AS400) remoto collegato via DRDA. Il nome va reperito col comando WRKRDBDIRE.
NomLib E' il nome della libreria in cui verranno effettuati gli accessi SQL se non si qualifica il nome file. (facoltativo).

NomFile E' il nome della file su cui verranno effettuati gli accessi se non si specifica la stringa SQL. (facoltativo).
Utente E' il nome dell'utente con cui verranno effettuati gli accessi SQL. (facoltativo).
Password E' la password corrispondente all'utente con cui verranno effettuati gli accessi SQL. (facoltativo).
DatiTesta E' una stringa che può contenere i dati esterni al subfile che si vogliono riportare nel foglio elettronico. Normalmente vi vengono passati i dati di testata che qualificano il contenuto delle righe. Ad esempio per un subfile di estratto conto come testata possono essere passati il nome del cliente e la data di aggiornamento dell'estratto conto stesso. Questo parametro può essere omesso. La sintassi all'interno della stringa è la seguente:
<nome campo>=XXX<valore>l
dove <nome campo video> è il nome del campo che verrà richiamato nella cella del foglio di calcolo con il nome fra caratteri %. XXX è la lunghezza del valore in caratteri. <valore> è la stringa da attribuire al campo. Il carattere di separazione l è il carattere pipe. La struttura può essere ripetuta più volte proseguendo subito dopo il carattere di separazione.
Ad esempio per passare la ragione sociale del cliente (RAGSOC) e la data di aggiornamento (DATAGG) la sintassi è la seguente:
RAGSOC=013PINCO PALLINOlDATAGG=01015/10/2005l
Grandi Estrazioni La possibilità di estrarre file in formato .xlsx viene risolta in 2 modalità, tramite questa flag è possibile indicare al convertitore che l’estrazione sarà molto grande (in termini di utilizzo di memoria) e quindi il file verrà scritto su disco man mano che procede l’operazione. Questa modalità è più veloce ma comporta alcuni limiti funzionali tra cui, no tabelle pivot, no grafici,no /QDLS come percorso di output.

Gli errori riportati dal comando si possono riferire alle seguenti aree:
Errore Parse XLS
- **RC 1** File Template non Esistente
- **RC 2** Errore I/O file Template
- **RC 3** File Template senza nessun foglio

Errore ParseDatiTesta XLS
- **RC 1** Errore Sintassi Dati Testata (Parametro DatiTesta)

Errore ApreFile AS
- **RC 1** Errore registrazione classe driver JDBC
- **RC 2** Errore in Connect
- **RC 3** Errore in Select -> Sostituito dall’errore SQLxxxx corrispondente
- **RC 4** Errore tipo driver

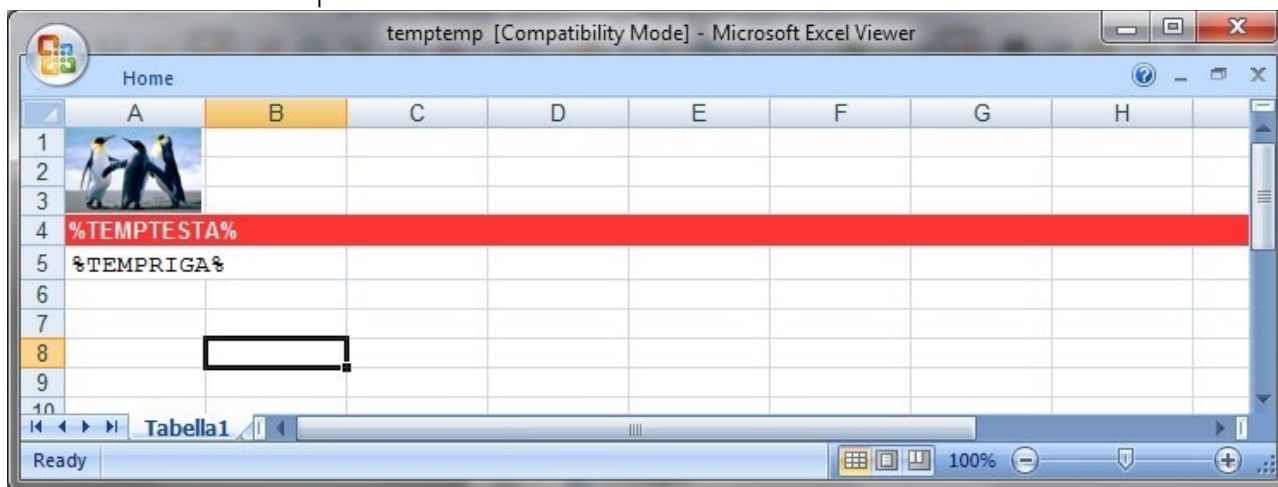
Errore ScriviTestata XLS
- **RC 1** Foglio Template non Esistente
- **RC 2** Errore reperimento righe Master
- **RC 3** Tipo Cella non previsto
- **RC 4** Errore lettura File AS
- **RC 5** Classe Cella non prevista
- **RC 8** Riga non creata, si è superato il numero max di righe

Errore ChiudeFile AS
- **RC 4** Errore chiusura File AS

Errore ScriveFile XLS

RC 1 Errore scrittura file risultante

Template generico

E' possibile configurare un template generico con il logo aziendale e le informazioni base tramite il template di template :
Creare una cartella nella cartella /NETACROSS/XLS con nome /template, all'interno della stessa aggiungere il file **temptemp.xls** (e/o .xlsx) con le specifiche minime di template.
Il campo **%TEMPTESTA%** indica la RIGA che verrà riempita con i nomi campi dell'estrazione in formato testo descrittivo come titolo della colonna, il campo **%TEMPRIGA%** indica la RIGA che verrà sostituita da i campi con **%NOMECAMPO%** che sarà la base dell'estrazioni XLS.

Template per utente

E' possibile differenziare il template per i subfile dei display, anche per utente: nella fase di creazione dell'xls se esiste nella struttura dei template sotto il livello LIBRERIA una cartella con NOME_UTENTE oppure una cartella GRUPPO_UTENTE, allora il file XLS risultante viene costruito sulla base del template personalizzato se corrisponde l'utente oppure il gruppo utente definito sulla configurazione profili di net-across

Profilo NETACROSS

Per poter eseguire correttamente l'estrazione, il job che esegue la conversione, deve avere un CCSID valido, cioè non può avere il CCSID 65535. Se gli utenti sono iscritti ad una validation list net-across verrà utilizzato il sottosistema netacross che ha definizione del job corretta.
E' possibile far eseguire i comandi java ad un solo utente definito con nome **NETACROSS** con CCSID impostato (italiano euro per esempio è 1144) e con autorizzazione all'uso del profilo da parte dell'utente QTMHHTP1, se così configurato allora il sottosistema utilizzerà questo profilo per eseguire i comandi java.
Da riga comandi as400:

```
EDTOBJAUT OBJ(NETACROSS) OBJTYPE(*USRPRF)
```

e impostare

```
QTMHHTP1         *USE
```

Nel caso che il valore CCSID sia 65535 si genererà un errore **ApreFile AS 3** in quanto la conversione EBCDIC non andrà a buon fine e l'SQL non sarà in grado di ritrovare i nomi campo

durante l’estrazione.

API Estrazione dati da un foglio di calcolo. (Cmd XLS2PF)

Così come è possibile da NetAcross creare fogli di calcolo partendo da archivi AS400 è possibile leggere fogli di calcolo per poterli elaborare da AS400.
Questo comando permette di estrarre il contenuto da un file PC nel formato foglio di calcolo (formato Excel’97 estensione .xls) all'interno di un file predefinito di NetAcross che rappresenta il contenuto delle celle non vuote.
[NEW 09/14] E’ possibile estrarre i dati anche da un file xlsx, !!ATTENZIONE i campi data in base alle diverse formattazioni possono essere riportati come data, numeri o stringhe!!
Un programma dell'utente può leggere il file predefinito riempito col contenuto del foglio di calcolo (una riga per ogni cella) e provvedere agli aggiornamenti del database AS400, ecc.
Il file PC può risiedere nel file system dell'AS400 oppure può risiedere su dei server esterni visibili da AS400 (supporto QNTC).

Per leggere da fogli di calcolo il comando è:

NETAP/XLS2PF

con i seguenti parametri:

Root E' il nome della directory di installazione NetAcross. Normalmente è '/NetAcross'. Deve essere inserita la barra iniziale e non deve essere inserita quella finale.
Path E' il percorso assoluto del foglio di calcolo. Se locale all'AS400 è il percorso IFS. Se esterno all'AS400 deve essere composto come /QNTC/<nome server>/<nome share>/<percorso>.
Libreria E' il nome della libreria in cui memorizzare il file predefinito per la memorizzazione dei fogli di calcolo. NetAcross fornisce un file di struttura di nome WXLSOUT nella libreria NETAP. Questo file viene copiato nella libreria indicata e poi viene riempito col contenuto del foglio di calcolo. Attenzione!: non si può usare la QTEMP.
[NEW 09/14] E’ disponilbe anche il parametro nome file:
File E' il nome del file in cui estrarre il contenuto del foglio di excel, i formato è quello di WXLSOUT ma è possibile indicare un nome file diverso.

La struttura del file WXLSOUT è la seguente:

WXNSHT	ZONAT.	5 0	Numero Foglio
WXNROW	ZONAT.	15 0	Numero Riga
WXNCOL	ZONAT.	15 0	Numero Colonna
WXTYPE	CHAR	10	Tipo Dato
WXSIZE	ZONAT.	10 0	Dimensione
WXDEC	ZONAT.	10 0	Caratteri Decimali
WXVALU	CHAR	256	Valore Stringa
WXFLGE	CHAR	1	Flag Errore

WXNSHT = numero di foglio contenuto nel file xls (da 1 a 256)
WXNROW = numero di riga nel foglio (da 1 a 65535 nel caso di xls)

WXNCOL = numero di colonna nella riga (da 1 a 256 nel caso di xls)
WXTYPE = tipo di dato, valori disponibili: 'STRING', 'NUMERIC', 'BOOLEAN', 'DATE'.
WXSIZE = numero di caratteri presenti nel campo valore, per i 'NUMERIC' è escluso il segno
WXDEC = solo per i 'NUMERIC' indica il numero di cifre decimali
WXVALUE = valore estratto dalla cella
Se il dato è STRING sarà inserito per la lunghezza WXSIZE.
Se il dato è NUMERIC avrà la formattazione: [+/-][cifre intere][cifredecimali]
WXSIZE=cifre intere+decimali. WXDEC=cifre decimali.
Se il dato è BOOLEAN sarà inserito per i valori true o false.
Se il dato è DATE avrà la formattazione 01-01-2001 00:00:00 dove le ore sono espresse nel formato 00-23.
WXFLGE = flag di errore, nel caso la cella contenga + caratteri di quelli estraibili conterrà 'E'. Se ok conterrà spazio.

Nel caso di celle con dati ricalcolati da formula, il dato nel file sarà del tipo e contenuto relativo alla formula calcolata.

API Invio di e-Mail. (Cmd AS2MAIL)

NetAcross prevede la possibilità di inviare e-Mail direttamente da iSeries.
Per inviare e-Mail il comando è:

NETAP/AS2MAIL

con i seguenti parametri:

Directory Root: E' il nome della directory di installazione NetAcross. Normalmente è '/NetAcross'. Deve essere inserita la barra iniziale e non deve essere inserita quella finale.
Nome Host SMTP: è il nome del server di posta a cui verrà inviata per la consegna la e-Mail oppure il server che fa il relay della e-Mail. Normalmente va indicato il nome del server interno di posta. Se indicato nella forma nomeserver:porta permette di specificare una porta di accesso al server SMTP diversa dal valore di default 25.
TO: indirizzo e-Mail della persona a cui si vuole mandare la e-Mail. Si possono indicare più nomi separati da virgola. (max 1024 caratteri)
Oggetto: l'oggetto della e-Mail. (max 256 caratteri)
Testo Corpo Messaggio: il testo del corpo della e-Mail. Si possono inserire i caratteri \n per andare a capo (inserisce CR/LF). (max 4096 caratteri)
E' possibile inserire del testo formattato HTML all'interno del corpo della mail, impostando ***HTML:** (asterisco HTML due punti) e dopo il testo html.
E' possibile inserire il contenuto di un file di testo (con formattazione html) impostando ***FILE:** (asterisco FILE due punti) e successivamente il percorso al file su IFS con path assoluto.
Files allegati: per allegare alla e-Mail dei files. I nomi devono essere separati dal carattere pipe (|). I files a cui si fa riferimento devono essere memorizzati nell'IFS dell'iSeries oppure referenziati ad altri server tramite il servizio NetServer. Il riferimento deve essere assoluto.

From: indirizzo e-Mail del mittente. In mancanza di questo l'iSeries crea automaticamente un mittente come <nome user as400>@<matricola as400>. Se il destinatario è interno all'azienda può funzionare, ma viene rifiutato se l'invio è destinato all'esterno. E' consigliato quindi riempire il campo.
Rispondere a: indica l'indirizzo e-Mail a cui potrà rispondere il destinatario.
CC: indirizzo e-Mail della persona a cui si vuoi mandare in copia conoscenza la e-Mail. Si possono indicare più nomi separati da virgola. (max 1024 caratteri)
BCC: indica l'indirizzo e-Mail della persona a cui si vuole mandare in copia conoscenza nascosta la e-Mail. Si possono indicare più nomi separati da virgola. (max 1024 caratteri)
Utente SMTP: se il server SMTP richiede autenticazione questo è l'utente con cui richiedere l'accesso. [NEW 09/2014 il parametro è lungo 120 char invece di 30 char]
Password SMTP: se il server SMTP richiede autenticazione questa è la password con cui richiedere l'accesso. [NEW 09/2014 il parametro è lungo 120 char invece di 30 char]
Conferma Consegna: chiede al server SMTP di restituire una mail di conferma consegna al server SMTP di destinazione. I valori possono essere:
*FAILURE ritorna una mail in caso di errori nella consegna (è il valore di default).
*SUCCESS ritorna una mail in caso di successo nella consegna.
*NEVER non ritorna mai nessuna mail.
*DELAY ritorna una mail in caso di ritardi nella consegna.
Ritorno per Conferma Consegna: se richiesta la conferma consegna specifica cosa deve tornare nella mail.
*HDRS ritorna una mail con la sola testata della mail originale.
*FULL ritorna una mail con la mail originale completa.
Ricevuta di Ritorno: chiede al server SMTP di restituire una mail di conferma lettura da parte dell'utente destinatario.
*NO non richiede la ricevuta di ritorno (è il valore di default).
*YES richiede la ricevuta di ritorno.

Comunicazione criptata e certificati

Nel caso si utilizzino certificati la JVM iSeries potrebbe non essere in grado di scaricare e accettare il certificato del servizio smtp in automatico. E' necessario quindi eseguire una procedura manuale per accettare il certificato e renderlo utilizzabile in fase di invio email.
Dalla riga comandi eseguire:

```
STRQSH
cd /netacross/as2mail
java -cp .:as2mail.jar as2mail.InstallCert nome_host_smtp[:porta]
```

A questo punto se il nome server è risolto correttamente e il certificato non è installato si dovrebbe ricevere un errore, e successivamente una lista di opzioni/certificati
Di solito è sufficiente premere 1 + Invio , cioè scegliere il primo proposto
Il certificato viene installato, ripetendo si ottiene

```
Starting SSL handshake...
No errors, certificate is already trusted
e successivamente di nuovo le opzioni di installazione.
```

Ricevuta di Ritorno: chiede al server SMTP di restituire una mail di conferma lettura da parte dell'utente destinatario.

*NO non richiede la ricevuta di ritorno (è il valore di default).
*YES richiede la ricevuta di ritorno.
Comunicazione criptata: permette di comunicare con un servizio tipo posta certificata con comunicazione protetta dei dati, la porta di default diventa la 425 se non diversamente specificato nel parametro server smtp.

API Creazione Documenti di Testo da Files di stampa. (Cmd SP2DOC)

NetAcross prevede la possibilità di creare documenti di testo PC da files di stampa direttamente da iSeries.
Per creare documenti di testo il comando è:

NETAP/SP2DOC

con i seguenti parametri:

Directory Root: E' il nome della directory di installazione NetAcross. Normalmente è '/NetAcross'. Deve essere inserita la barra iniziale e non deve essere inserita quella finale.
Percorso Finale DOC E' il nome del documento di testo che verrà creato come risultato finale. Il file risiederà in una directory del IFS AS400. Attenzione: se esistente verrà sostituito.
Nome Lavoro: è il nome del lavoro che ha creato il file di stampa.
Utente: è il nome dell'utente che ha creato il file di stampa.
Numero: è il numero del lavoro che ha creato il file di stampa.
File in spool: è il nome del file che ha creato il file di stampa.
Numero File in spool: è il numero del file che ha creato il file di stampa.

I 5 parametri precedenti identificano il file di stampa da trasferire in documento di testo. Il file di stampa deve essere in formato SCS. Il formato prodotto non è compatibile con OpenOffice.

Gli errori riportati dal comando si possono riferire alle seguenti aree:
Errore Parse DOC
RC 1 File Template non Esistente
RC 2 Errore I/O file Template
RC 3 Non trovato segnalibro %SPOOLFILE%
Errore ScriviFile DOC
RC 1 Errore scrittura file risultante

API Creazione Documenti in formato PDF da Files di stampa. (Cmd SP2PDF)

NetAcross prevede la possibilità di creare documenti in formato PDF da files di stampa direttamente da iSeries.
Per creare documenti in formato PDF il comando è:

NETAP/SP2PDF

con i seguenti parametri:

Directory Root: E' il nome della directory di installazione NetAcross. Normalmente è '/NetAcross'. Deve essere inserita la barra iniziale e non deve essere inserita quella finale.
Percorso Finale PDF E' il nome del documento in formato PDF che verrà creato come risultato finale. Il file risiederà in una directory del IFS AS400. Attenzione: se esistente verrà sostituito.
Percorso Regole PDF E' il nome di un file in formato XML che deve contenere le regole per la creazione del documento PDF. Il file deve risiedere in una directory del IFS AS400. Se non esiste vengono usate delle regole standard in funzione della larghezza della pagina.
Nome Lavoro: è il nome del lavoro che ha creato il file di stampa.
Utente: è il nome dell'utente che ha creato il file di stampa.
Numero: è il numero del lavoro che ha creato il file di stampa.
File in spool: è il nome del file che ha creato il file di stampa.
Numero File in spool: è il numero del file che ha creato il file di stampa.

I 5 parametri precedenti identificano il file di stampa da trasferire in documento di testo. Il file di stampa deve essere in formato SCS.
Un esempio di file delle regole PDF è il seguente:

```
<?xml version="1.0" encoding="UTF-8"?>
<SP2PDF>
        <Page>
                <Size>A4</Size>
                <Orientation>0</Orientation>
        </Page>
        <Border>
                <Left>30</Left>
                <Right>30</Right>
                <Top>20</Top>
                <Bottom>20</Bottom>
        </Border>
        <Font>
                <Name>Courier</Name>
                <Size>11</Size>
                <AddVert>-5</AddVert>
        </Font>
        <WaterMark>
                <Name></Name>
        </WaterMark>
</SP2PDF>
```

Il significato delle singole chiavi è:
<SP2PDF> chiave base che comprende tutto il documento XML.
<Page> chiave che racchiude le dimensioni della pagina.
<Size> chiave che identifica la dimensione della pagina. Valori validi A0, A1, A2, A3, A4. Parametro obbligatorio.
<Orientation> chiave che identifica l'orientamento della pagina. Valori validi 0, 90, 180, 270. Se non espresso vale 0 (ritratto verticale).

<Border> chiave che racchiude la dimensione dei bordi della pagina.
<Left> chiave che identifica il margine sinistro in punti per pollice. Se non espresso vale 0.
<Right> chiave che identifica il margine destro in punti per pollice. Se non espresso vale 0.
<Top> chiave che identifica il margine alto in punti per pollice. Se non espresso vale 0.
<Bottom> chiave che identifica il margine basso in punti per pollice. Se non espresso vale 0.
<Font> chiave che racchiude lo stile dei paragrafi scritti.
<Name> chiave che identifica il nome del font da utilizzare. Parametro obbligatorio.
<Size> chiave che identifica l'ampiezza del font da utilizzare. Parametro obbligatorio.
<AddVert> chiave che identifica il numero di punti da aggiungere alla spaziatura standard verticale del font. E' possibile inserire anche i decimali (utilizzando il punto come virgola) ed il segno (che deve precedere il valore). Parametro obbligatorio.
<WaterMark> chiave che racchiude la sovrapposizione immagine.
<Name> chiave che identifica il nome dell'immagine da sovrapporre ad ogni pagina del file in formato PDF generato. Il file deve risiedere in una directory del IFS AS400. Deve essere in un formato PC e verrà scalata all'ampiezza della pagina. Parametro facoltativo.

E' disponibile il font "Unicode.ttf" che può essere utilizzato per rappresentare caratteri non iso-latin, impostando il tag <Font>Unicode</Font> e avendo cura di copiare il file Unicode.ttf nella stessa cartella del file xml di configurazione utilizzato.

Gli errori riportati dal comando si possono riferire alle seguenti aree:

Errore Parse XML

RC 1 Errore creazione istanza Reader
RC 2 Errore I/O file XML
RC 3 Errore sintassi file XML

Errore crea PDF

RC 1 Manca sigla ampiezza pagina (A0, A1, A2, A3, A4)
RC 2 Orientamento diverso da 0, 90, 180, 270.
RC 3 Errore documento file PDF
RC 4 Errore I/O file PDF
RC 5 Watermark – Elemento errato
RC 6 Watermark – URL errato
RC 7 Watermark – Errore I/O
RC 8 Font – Non trovato

Errore insertRiga PDF

RC 1 Errore documento

API Recupero contenuto di un fil email

Esiste una funzione molto particolare che permette di elaborare un file .eml (cioè una email) se questo è depositato su IFS o comunque raggiungibile tramite un path condiviso da as400, la funzione è richiamabile tramite comando soltanto da programma CL:

NETAP/EML2AS

con i seguenti parametri:

Directory Root ROOT: E' il nome della directory di installazione NetAcross. Normalmente è '/NetAcross'. Deve essere inserita la barra iniziale e non deve essere inserita quella finale.
Percorso Finale .eml FILEEML: E' il percorso assoluto dove trovare il file email da elaborare
Attach Output DIRPATH: è il nome della cartella dove verrà esploso il contenuto di allegati della mail
FROM: ritorna il parametro FROM della mail
RISPA: ritorna il parametro RISPA della mail
TO: ritorna il parametro TO della mail
SENT: ritorna la data di invio della mail
SUBJECT: ritorna il parametro SUJECT della mail
MESSAGE: ritorna il corpo del messaggio in formato testo
NOMEFILE: la lista dei nomi file allegati separati dal carattere pipe "|"
CC: ritorna il parametro CC della mail
BCC: ritorna il parametro BCC della mail

Il comando è richiamabile da CL o RPG ma non da riga comandi, la cartella dove verranno esplosi gli allegati deve esistere e non verrà creata, la lista degli allegati contiene i nomi file dei documenti estratti nella cartella indicata.

API Creazione Grafici. (cmd AS2CHART)

NetAcross prevede la possibilità di creare grafici in formato immagine PC direttamente da iSeries sulla base di dati forniti dall'applicazione utente. I grafici prodotti possono poi essere collegati alle pagine dell'applicazione con le normali keyword NetAcross (*LNK, *IML, *IMG, *EVT).
Per creare grafici in formato immagine PC il comando è:

NETAP/AS2CHART

con i seguenti parametri:

Directory Root: E' il nome della directory di installazione NetAcross. Normalmente è '/NetAcross'. Deve essere inserita la barra iniziale e non deve essere inserita quella finale.
Percorso Dati Chart E' il nome di un file in formato XML che deve contenere le regole ed i dati per la creazione del grafico. Il file deve risiedere in una directory del IFS AS400.
Percorso Destinazione Immagine Chart E' il nome dell'immagine in formato PC che verrà creato come risultato finale. Il file risiederà in una directory del IFS AS400. Attenzione: se esistente verrà sostituito. Se l'immagine verrà visualizzata tramite Alias anche quest'ultimo dovrà essere creato.

Un esempio di dati Chart è il seguente:

```
<?xml version="1.0"?>
<AS2CHART>
  <Chart>
    <Titolo>Vendite Italia</Titolo>
    <Legend>true</Legend>
```

```
    <ToolTips>true</ToolTips>
    <Urls>true</Urls>
    <Tipo>Pie3D</Tipo>
  </Chart>
  <PieDataSet>
    <Serie>
      <ValoreX>Nord Italia</ValoreX>
      <ValoreY>1234000</ValoreY>
      <ValoreX>Centro Italia</ValoreX>
      <ValoreY>1072100</ValoreY>
      <ValoreX>Sud Italia</ValoreX>
      <ValoreY>2533300</ValoreY>
      <ValoreX>Isole</ValoreX>
      <ValoreY>511200</ValoreY>
    </Serie>
  </PieDataSet>
  </AS2CHART>
```

Il significato delle singole chiavi è:

<AS2CHART> chiave base che comprende tutto il documento XML.
<Chart> chiave che racchiude i dati generali del grafico.
<Height> altezza in pixel del grafico generato
<Width> larghezza in pixel del grafico generato
<Titolo> chiave che identifica il titolo del grafico.
<Legend> chiave che identifica se inserire la legenda dei valori. (true o false)
<ToolTips> chiave che identifica se includere la decodifica dei singoli valori.(true o false)
<Urls> chiave che identifica se includere la decodifica dei valori.(true o false)
<PercentDiffForMaxScale> (solo PieDiff) identifica se evidenziare la percentuale massima. Inserire il valore numerico di soglia.
<GreenForIncrease> (solo PieDiff) identifica se evidenziare in verde gli incrementi. (true o false)
<SubTitle> chiave specifica per il grafico PieDiff che identifica il sottotitolo.
<ShowDifference> chiave che identifica se evidenziare le differenze. (true o false)
<Ordine> identifica quale asse utilizzare per l'identificazione del grafico.
<Tipo> chiave che identifica il tipo di grafico richiesto. I valori possibili sono elencati nella tabella che segue.

<PieDataSet> chiave che racchiude i dati dei grafici di tipo torta.
<Serie> chiave che racchiude i dati del grafico.
<ValoreX> chiave che identifica il valore da esprimere sull'asse delle ascisse.
<ValoreY> chiave che identifica il valore da esprimere sull'asse delle ordinate.

<XYDataSet> chiave che racchiude i dati dei grafici di tipo X/Y.
<Serie> chiave che racchiude i dati del grafico.
<AsseX> chiave che identifica la descrizione dell'asse delle ascisse.
<AsseY> chiave che identifica la descrizione dell'asse delle ordinate.
<PlotOrientation> identifica l'orientamento del grafico(V=Vertiicale O=Orizzontale)
<Titolo> chiave che identifica la descrizione della parte dati del grafico.
<ValoreX> chiave che identifica il valore da esprimere sull'asse delle ascisse.
<ValoreY> chiave che identifica il valore da esprimere sull'asse delle ordinate.

<CategDataSet> chiave che racchiude i dati dei grafici di tipo categoria.

<Serie> chiave che racchiude i dati del grafico.
<AsseX> chiave che identifica la descrizione dell'asse delle ascisse.
<AsseY> chiave che identifica la descrizione dell'asse delle ordinate.
<PlotOrientation>identifica l'orientamento del grafico (V=Verticale O=Orizzontale)
<ValoreX> chiave che identifica il valore da esprimere sull'asse delle ascisse.
<ValoreY> chiave che identifica il valore da esprimere sull'asse delle ordinate.
<ValoreZ> chiave che identifica il valore da esprimere sul terzo asse.
<RangeMin> valore minimo rappresentato per l’asse dei valori (solo Line)
<RangeMax> valore massimo rappresentato per l’asse dei (solo Line)

<IntervalDataSet> chiave che racchiude i dati dei grafici di tipo intervallo.
<Serie> chiave che racchiude i dati del grafico.
<Titolo> chiave che identifica la descrizione della parte dati del grafico.
<Livello> chiave che identifica il livello del lavoro. Numerico.
<Task> chiave che identifica la descrizione del lavoro.
<DataI> chiave che identifica la data inizio del lavoro. Nella forma GGMMAAAA.
<DataF> chiave che identifica la data fine del lavoro. Nella forma GGMMAAAA.
<Perc> chiave che identifica la percentuale di avanzamento del lavoro. Numerico.

<IntervalXYDataSet> chiave che racchiude i dati dei grafici di tipo intervallo X/Y.
<Titolo> chiave che identifica la descrizione della parte dati del grafico.
<Serie> chiave che racchiude i dati del grafico.
<Data> chiave che identifica la data inizio del lavoro. Nella forma GGMMAAAA.
<Valore> chiave che identifica l'avanzamento del lavoro. Numerico.

Le chiavi sono valide in funzione del contesto di grafico a cui si riferiscono. La tabella che segue evidenzia la validità specifica.

Chart	Titolo	AsseX	AsseY	XYTitolo	PieDataSet/Serie	Orientation	ToolTips	Urls	PercentDiffForMaxScale	GreenForIncrease	SubTitle	ShowDifference	Ordine
Pie	X				PieDataSet ValoreX/ValoreY		X	X					
Ring	X				PieDataSet ValoreX/ValoreY		X	X					
Pie3D	X				PieDataSet ValoreX/ValoreY		X	X					
PieDiff	X				PieDataSet ValoreX/ValoreY*2		X	X	X	X	X	X	
MultiplePie	X				CategDataset ValoreX/ValoreY/ValoreZ		X	X					Col./Row
MultiplePie3D	X				CategDataSet ValoreX/ValoreY/ValoreZ		X	X					Col./Row
Bar	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X					
Bar3D	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X					
StackedBar	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X					
StackedBar3D	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X					

Area	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X
StackedArea	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X
Line	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X
Line3D	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X
Waterfall	X	X	X		CategDataSet ValoreX/ValoreY/ValoreZ	V/O	X	X
Polar	X			X	XYDataSet ValoreX/ValoreY		X	X
ScatterPlot	X	X	X	X	XYDataSet ValoreX/ValoreY	V/O	X	X
XYArea	X	X	X	X	XYDataSet ValoreX/ValoreY	V/O	X	X
XYLine	X	X	X	X	XYDataSet ValoreX/ValoreY	V/O	X	X
XYStep	X	X	X	X	XYDataSet ValoreX/ValoreY	V/O	X	X
XYStepArea	X	X	X	X	XYDataSet ValoreX/ValoreY	V/O	X	X
TimeSeries	X	X	X	X	XYDataSet ValoreX/ValoreY		X	X
Gantt	X	X	X		IntervalCategDataSet Livello/Task/DataI/DataF/Perc		X	X
XYBar	X	X	X		IntervalXYDataSet ValoreX/ValoreY	X	X	X
Histogram	X	X	X		IntervalXYDataSet ValoreX/ValoreY	X	X	X

Gli errori riportati dal comando si possono riferire alle seguenti aree:

Errore Parse XML

RC 1 Errore creazione istanza Reader

RC 2 Errore I/O file XML

RC 3 Errore sintassi file XML

Errore Crea Chart

Errore Sintassi parametri Chart

Errore Scrive Chart

RC 1 Errore scrittura grafico.

Per facilitare la creazione del file XML da normali programmi procedurali viene fornito insieme a NetAcross il programma di servizio **UTILXML**. L'utilizzo essendo un programma di servizio è consentito solo ai programmi in ambiente ILE. E' disponibile la Bind Directory **CHARTXML** nella libreria **NETAP** per bindare la compilazione del programma utente.

Sono diponibili le seguenti procedure:

```
CreaXML          PR              10I 0
 prPath                         256     Value
 prVersione                      10     Value
```

da eseguire la prima volta fornendo il percorso dove creare il file XML e la versione XML (normalmente 1.0). Il valore di ritorno è 0 se eseguito correttamente. Altrimenti è il codice di errore decodificabile con DSPMSGD CPEXXXX.

```
ApriKwd          PR              10I 0
 prKwd                          256     Value
 prParmKwd                      256     Value
 prDatiKwd                      256     Value
```

da eseguire per aprire una keyword XML. Fornire il nome della keyword, eventuali parametri della keyword e i suoi dati. I valori sono delimitati dall'ultimo spazio a destra. Possono essere aperte keyword nidificate fra loro. Il valore di ritorno è 0 se eseguito correttamente. Altrimenti è il codice di errore decodificabile con DSPMSGD CPEXXXX.

```
ChiudiKwd        PR              10I 0
 prKwd                          256     Value
```

da eseguire per chiudere una keyword XML. Fornire il nome della keyword da chiudere. Il valore di ritorno è 0 se eseguito correttamente. Il valore di ritorno è 1 se si sta chiudendo una chiave mai aperta. Altrimenti è il codice di errore decodificabile con DSPMSGD CPEXXXX.

```
ChiudiXML        PR              10I 0
```

da eseguire per chiudere il file XML. Eventuali chiavi aperte vengono chiuse automaticamente. Il valore di ritorno è 0 se eseguito correttamente. Altrimenti è il codice di errore decodificabile con DSPMSGD CPEXXXX.

API Gestione VLDL

Tramite queste API è possibile manipolare a run-time il contenuto di una validation list presente in NETAF, inserendo, modificando o eliminando gli utenti iscritti.

ADDVLDL : Aggiunta di un utente in validation list

CHGVLDL : Cambio informazioni si utente in validation list

Entrambe hanno la stessa interfaccia.

```
C      *Entry         PList
C                     Parm                     RC            10 0
```

```
C                   Parm                    Validation    10
C                   Parm                    UtenteWeb     30
C                   Parm                    Password      10
C                   Parm                    TipoUtente     3
C                   Parm                    Codice         6
C                   Parm                    Descrizione   30
C                   Parm                    UserAs400     10
C                   Parm                    Terminale     10
C                   Parm                    Env            3
C                   Parm                    UserAsPwd     10
C                   Parm                    LoginMenu     10
C                   Parm                    LoginPgm      10
C                   Parm                    AuthLvl        2
C                   Parm                    AbilGzip       1
C                   Parm                    UtilityObj    20
C                   Parm                    GruppoUser    10
C                   Parm                    CssFile       30
C                   Parm                    Appl          10
C                   Parm                    Pers          10
C                   Parm                    Lang           3
C                   Parm                    Date          10
 *
```

DELVLDL: Eliminazione utente da validation list

```
 *
C     *Entry        PList
C                   Parm                    RC               10 0
C                   Parm                    Validation       10
C                   Parm                    UtenteWeb        30
 *
```

I parametri hanno questo significato:

RC	10,0	Deve essere passato a 0 e ritorna un codice errore (vedi tabella errori a seguire)
Validation	10	Nome della VLDL così come è presente nella libreria NETAF
UtenteWeb	30	Nome utente web
Password	10	Password utente web
TipoUtente	3	Tipo utente come presente nella configurazione del profilo del menù gestione utenti net-across
Codice	6	Codice applicativo associato all'utente come presente nella configurazione del profilo del menù gestione utenti net-across
Descrizione	30	Descrizione libera dell'utente, informazioni aggiuntive del profilo utente
UserAs400	10	Utente di sistema operativo usato dai job a s400
Terminale	10	Nome completo o radice parziale usato per identificare i job NETVT

Env	3	Ambiente di utilizzo INT o BCH
UserAsPwd	10	Password per utente as400 se usato in interattivo (ambiente INT)
LoginMenu	10	Menu di login (sostituisce il menù al momento della connessione) solo INT
LoginPgm	10	Programma di login (sostituisce il programma nella login di connessione) solo INT
AuthLvl	2	Valore di 2 cifre indicante un livello di autorizzazione logico sul profilo web
AbilGzip	1	Abilita la compressione gzip delle pagine html trasmesse
UtilityObj	20	Nome del gruppo funzioni estese di Utilità
GruppoUser	10	Gruppo utenti logici per utenti web
CssFile	30	Nome del file css usato per renderizzare le pagine
Appl	10	Nome Applicazione di riferimento
Pers	10	Nome Personalizzazione di riferimento
Lang	3	Lingua impostata sul profilo
Date	10	Data formato ISO di scadenza password (01-01-0001 per indicare la password scaduta e quindi richiedere il cambio alla prossima login)

Tabella Errori API VLDL

Codice Errore	Descrizione	Valori validi
60	Utente non aggiunto, problema non specificato, controllare le log	

EXIT

EXIT per riempimento dati keyword DDS SEV

La keyword DDS SEV può indicare in nome di una exit che deve fornire i dati per il riempimento della select box che la keyword crea. Questa exit è indicata per il riempimento di caselle di selezione sulla base di tabelle variabili dell'applicazione.
Per l'utilizzo vedere la keyword *SEV nel capitolo Direttive Html nelle DDS in questo manuale.

Il programma di exit dovrà definire i seguenti parametri:
Valore dinamico (input 256 alfabetico): viene riempito con il risultato del parametro della keyword (**v:**) . E' utile quando il programma di exit deve generare la lista personalizzandola in funzione del valore di alcuni campi della pagina video. Altrimenti può essere lasciato vuoto.
Nome campo (input 10 alfabetico): viene riempito con il nome del campo su cui è abilitata la keyword.
Lunghezza campo (input 10 numerico packed): viene riempito con la lunghezza del campo su cui è abilitata la keyword.
Lunghezza descrizione (output 10 numerico packed): il programma di exit deve riempire questo campo con la lunghezza della descrizione che verrà ritornata nel parametro elementi. Il valore si riferisce alla singola descrizione.
Numero elementi (output 10 numerico packed): il programma di exit deve riempire questo campo con il numero di descrizioni che verranno ritornati nel parametro elementi.
Elementi (output 8000 alfabetico): il programma di exit deve riempire questo campo con il numero di descrizioni previsto nel parametro Numero Elementi, con ogni valore lungo come previsto nel parametro Lunghezza Descrizione. Ogni descrizione deve essere adiacente all'altra, compresi gli spazi.
Il programma di exit avrà già impostata la lista librerie corretta. Poiché tale programma verrà eseguito nel lavoro principale occorrerà porre attenzione dell'esecuzione di operazioni particolari (Override, variazioni lista librerie, ecc.) che potrebbero avere ripercussioni sul chiamante.

EXIT per validazione richieste di HyperLink

Nella gestione Alias/Directory del menu Configurazione è possibile indicare un programma che verifica sulla base dei parametri ricevuti se il link richiesto deve essere autorizzato oppure no.
Il programma deve ricevere i seguenti parametri:

RemoteUser è il nome utente di collegamento. Parametro 1: 30 caratteri alfabetico.
As400User è il nome utente di riferimento iSeries. Parametro 2: 10 caratteri alfabetico.
IDn è il numero sessione di collegamento. Parametro 3: 11 numerico tipo S 0 decimali.
File è il nome file richiesto dall'hyperlink. Parametro 4: 256 caratteri alfabetico.
Path è il nome della directory iSeries dove cercare il file richiesto dall'hyperlink. Parametro 5: 256 caratteri alfabetico.

VldlList è il nome della validation list che ha validato l'utente (vuoto se utente di sistema). Parametro 6: 10 caratteri alfabetico.
UsrType è il tipo dell'utente che ha richiesto il link (vuoto se utente di sistema). Parametro 8: 3 caratteri alfabetico. Valido CLI/FOR/AGE oppure vuoto.
UsrCode è il codice anagrafico dell'utente che ha richiesto il link (vuoto se utente di sistema). Parametro 9: 6 caratteri alfabetico.
UsrDesc è la descrizione dell'utente che ha richiesto il link (vuoto se utente di sistema). Parametro 10: 30 caratteri alfabetico.

Il programma deve restituire un parametro:

RC return code. Parametro 7: 1 numerico tipo S 0 decimali con il valore:
0 (true) = autorizzato all'accesso.
-1 (false) = Non autorizzato.

EXIT Impostazione parametri di Login

Quando un utente si collega a NetAcross eredita dall'utente i valori del menù e/o del programma iniziale. Questa exit permette se abilitata di forzare questi parametri sulla base di scelte fatte partendo da tabelle, ecc. Permette inoltre di ricevere un parametro libero nell'URL e utilizzarlo per logiche personalizzate. Vedi NETAPARMP in questo manuale.

Configurazione

<FormLoginExit>*nome_libreria/nome_programma***</FormLoginExit>**: per abilitare la funzionalità descritta è necessario indicare il nome del programma che si occupa delle operazioni di login

NAPgmCaller è il nome del programma che sta ricevendo la login. Può valere NET_LOGIN oppure NET_ACROS2. Parametro 1: 10 caratteri alfabetico.
NAUser è il nome utente di collegamento. Parametro 2: 30 caratteri alfabetico.
NAPwd è la password dell'utente di collegamento. Parametro 3: 10 caratteri alfabetico.
NASis è il sistema informativo dell'utente di collegamento. Parametro 4: 10 caratteri alfabetico.
NAAzn è l'azione dell'utente di collegamento. Parametro 5: 10 caratteri alfabetico.
NAUrl è l'URL di ritorno da eseguire al termine dell'azione. Parametro 6: 256 caratteri alfabetico.
NAPrm1 sono i parametri passati nella login. Parametro 7: 132 caratteri alfabetico.

EXIT Menù ad albero con architetture non standard

Queste exit vengono richiamate da NetAcross in base a impostazioni del file Config.xml. Servono a definire il riempimento ed il funzionamento del menù ad albero per architetture non standard (diverse cioè da ACG, SAM e Sistema Operativo).

Insieme al prodotto nei file QCLLESRC e QRPGLESRC nella libreria NETAP vengono forniti i sorgenti delle exit relative al modulo base ACG, su cui sarà facile per l'utente effettuare le modifiche necessarie alle proprie esigenze. Sarà opportuno portare i sorgenti in proprie librerie e dopo le modifiche opportune compilarli sempre su proprie librerie. Poi sarà necessario modificare il file Config.xml duplicando il gruppo di chiavi <ArcDef> ACG, chiamandolo con un nome proprio e modificando il nome delle exit da richiamare puntando alla propria libreria ed eventualmente modificando anche gli altri dati della struttura Xml.
La exit di ACG, fornita di default, prevede la seguente configurazione nel file Config.xml:

```
<ArcList>
 <ArcDef>
  <ArcName>ACG</ArcName>
  <ArcPgm>ACGGAA/BCI00</ArcPgm>
  <ArcUsrNfo>NETAP/NAUSRINFAC</ArcUsrNfo>
  <ArcMnuLst>NETAP/NASRVMNACP</ArcMnuLst>
  <ArcSisLst>NETAP/NASISINFA</ArcSisLst>
  <ArcVidMnu>NETAP/BMAMN00V.BMAMN01V</ArcVidMnu>
  <ArcHomKey>F16</ArcHomKey>
  <ArcFldCho>SCELTA</ArcFldCho>
  <ArcPreSbm>NETAP/NABCH10_AC</ArcPreSbm>
  <ArcPreCal>NETAP/PRECALLACG</ArcPreCal>
  <ArcPosCal></ArcPosCal>
  <ArcPreEst>NETAP/NABCH2BAP</ArcPreEst>
 </ArcDef>
</ArcList>
```

Il parametro **ArcName** è descrittivo e identifica l'architettura a cui si fa riferimento.

Il parametro **ArcPgm** indica come abbinare questo tipo di architettura all'utente che si collega. Se l'User Profile che si sta collegando ha nel parametro INLPGM il programma specificato. Se avviene la corrispondenza viene attivato questo gruppo di Exit.

Il parametro **ArcUsrNfo** indica il programma da eseguire per ottenere le informazioni relative all'utente collegato. Vengono passati i seguenti parametri:

NomeUtente	10A	IN	Nome Utente (UsrPrf)
IniPgmName	10A	OUT	Nome Pgm iniziale utente di architettura
IniPgmLibr	10A	OUT	Libr.Pgm iniziale utente di architettura
EndPgmName	10A	OUT	Non usato
EndPgmLibr	10A	OUT	Non usato
LibArch	10A	OUT	Libreria dei file di architettura
MultSysInf	1A	OUT	Abilitati più sistemi informativi (S/N)
SysInfX	10A	OUT	Nome sistema informativo iniziale
MnuIniX	10A	OUT	Nome menù iniziale
TreeMenuInt	1A	OUT	Attivazione menù ad albero (Y/N)
RtnCode	2A	OUT	Return Code (Attualmente non usato)

			Impostare a Blank

Per questa exit vengono forniti i sorgenti per l'architettura ACG: NAUsrInfAC (RPGLE) e NAUsrInfAP (CLLE).

Il parametro **ArcMnuLst** indica il programma da eseguire per ottenere le informazioni relative alla catena di menù assegnata all'utente specifico. Vengono passati i seguenti parametri:

NomeUtente	10A	IN	Nome Utente (UsrPrf)
Env	3A	IN	Modalità NetAcross (INT/BCH)
SysInf	10A	IN	Nome sistema informativo iniziale
MnuIni	10A	IN	Menù iniziale utente (può essere vuoto)
Sep	1A	IN	Carattere separatore della lista
pHtmlOut	*	IN	Puntatore al buffer della lista
HtmlDim	9B0	IN	Dimensione in bytes massima della lista
HtmlCur	9B0	OUT	Dimensione in bytes effettiva della lista
RtnCode	2A	OUT	Return Code (Attualmente non usato) Impostare a Blank

Per questa exit vengono forniti i sorgenti per l'architettura ACG: NASrvMnACP (CLLE) e NASrvMnAC (RPGLE).
Il puntatore pHtmlOut indica un'area di memoria disponibile a contenere i caratteri di ritorno. Il formato di risposta sarà un elenco con questa sequenza:
LivelloProfondità(2A)+SEP+NumeroAzione(2A)+SEP+TipoAzione(1A)+SEP+Descrizione+CRLF
SEP è il Separatore passato dal chiamante
CRLF sono 2 caratteri carriage return + line feed
LivelloProfondità: codice con valore da 01 a 99 che indica il livello di profondità del ramo dell'albero dei menù.
NumeroAzione: codice con valore da 01 a 99 (per ACG da 01 a 14) che indica l'opzione rispetto al menù visualizzato.
TipoAzione: codice di 1 alfabetico con valore 'A' per un azione e 'M' per un menù.
Descrizione: testo descrittivo dell'azione o del menù, visualizzato nell'albero.
Per esempio se separatore è la virgola ',' il risultato sarà:
01,01,A,Gestione Ordini
01,02,M,Menù Stampa Ordini
02,01,A,Stampa Ordini per Cliente
02,02,A,Stampa Ordini per Agente

Il parametro **ArcSisLst** indica il programma da eseguire per ottenere le informazioni relative all'elenco dei sistemi informativi disponibili per l'utente specifico. Vengono passati i seguenti parametri:

NomeUtente	10A	IN	Nome Utente (UsrPrf)
MnuIni	10A	IN	Menù iniziale utente (può essere vuoto)
Sep	1A	IN	Carattere separatore della lista
pHtmlOut	*	IN	Puntatore al buffer della lista

HtmlDim	9B0	IN	Dimensione in bytes massima della lista
HtmlCur	9B0	OUT	Dimensione in bytes effettiva della lista
RtnCode	2A	OUT	Return Code (Attualmente non usato) Impostare a Blank

Per questa exit vengono forniti i sorgenti per l'architettura ACG: NASisInfA (RPGLE).
Il puntatore pHtmlOut indica un'area di memoria disponibile a contenere i caratteri di ritorno. Il formato di risposta sarà un elenco con questa sequenza:
CodiceSistemaInf. + SEP + DescrizioneSistemaInformativo + SEP + MenùIniziale+ CRLF
SEP è il Separatore passato dal chiamante
CRLF sono 2 caratteri carriage return + line feed
Per esempio se separatore è la virgola ',' il risultato sarà:
SISTEMA01,Sistema Test 01,M001
SISTEMA02,Sistema Test 02,M002

Il parametro **ArcVidMnu** è necessario solo nella modalità NetAcross interattiva. Indica il nome del file video che presenta l'interfaccia per i menù e le azioni. Si esprime nel formato completo Libreria/File.Formato con un massimo di 32 caratteri.

Il parametro **ArcHomKey** è necessario solo nella modalità NetAcross interattiva. Indica il tasto funzionale del file video precedentemente specificato nella ArcVidMnu che esegue il riposizionamento a menù iniziale. Può avere valore “Home” o un valore tra “F01” e “F24”

Il parametro **ArcFldCho** è necessario solo nella modalità NetAcross interattiva. Indica il campo del file video precedentemente specificato nella ArcVidMnu che riceve le scelte da parte dell'utente.

Il parametro **ArcPreSbm** è necessario solo nella modalità NetAcross batch. Indica il programma da eseguire per ottenere le informazioni relative all'esecuzione dell'azione che verrà eseguita in base alla selezione dal menù ad albero. Vengono passati i seguenti parametri:

NomeUtente	10A	IN	Nome dell'utente (UsrPrf)
NomeJob	10A	IN	Nome del job (terminale virtuale) della sessione utente.
SisInf	10A	IN	Sistema informativo attivo.
MnuIni	10A	IN	Menù iniziale attivo.
SeqAzn	100A	IN	Sequenza azione fatta dall'utente
LibArch	10A	IN	Libreria di architettura
MultiSisInf	1A	IN	Abilitati più sistemi informativi (S/N)
FlagAuto	1A	OUT	Flag azione int. o batch (KAUTO per ACG) S=Az.Batch N=Az.Interattiva
FlagCfSc	1A	I/O	Flag di scambio per conferma (S/N/K).
CmdExec	1024A	OUT	Dati CALL + parametri per esecuzione.
ProgAutor	1A	OUT	Flag autorità azione livello QPgmr (S/N).
JobQueue	21A	OUT	Parametro JobQ per SbmJob (libr/ogg). Compilare con la barra e senza spazi.
JobDescr	21A	OUT	Parametro JobD per SbmJob (libr/ogg).

			Compilare con la barra e senza spazi.
OutQueue	21A	OUT	Parametro OutQ per SbmJob (libr/ogg). Compilare con la barra e senza spazi.
RtgDta	5A	OUT	Parametro RtgDta per SbmJob. Normalmente QCMDB per batch e QCMDI per interattivo.
CurLib	10A	OUT	Parametro CurLib per SbmJob.
LibList	2750A	OUT	Parametro InLibL per SbmJob.
RtnCode	2A	OUT	Valore di ritorno (blank o diverso).

Il campo FlagCfSc indica se l'azione da eseguire prevede conferma e prevede due passaggi nel caso di 'S' confermando con 'K' .
Il campo RtnCode nel caso sia diverso da *blanks, il chiamante non esegue la sbmjob (nel caso di ACG per azioni non tipo PG è l'exit che esegue la SBMJOB).
Per questa exit vengono forniti i sorgenti per l'architettura ACG: NABch10_AC (CLLE),NABch10BA (RPGLE), NAKPJBA (RPGLE), NARtvAzA (RPGLE) e NARtvPrA (RPGLE).

Il parametro **ArcPreCal** è necessario solo nella modalità NetAcross batch. Indica il programma da eseguire prima dell'esecuzione dell'azione selezionata dall'albero dei menù. Riceve il comando di lancio azione per il sistema operativo ed ha la possibilità di modificarlo. Vengono passati i seguenti parametri:

CmdExec	1024A	I/O	Comando di lancio azione per il S.O.
RtnCode	2A	OUT	Return Code (Attualmente non usato) Impostare a Blank

Per questa exit vengono forniti i sorgenti per l'architettura ACG: PreCallACG (CLLE), NAInrAzA (RPGLE).

Il parametro **ArcPosCal** è necessario solo nella modalità NetAcross batch. Indica il programma da eseguire dopo l'esecuzione dell'azione selezionata dall'albero dei menù. Riceve il comando di lancio azione. Questa Exit attualmente non è implementata. Vengono passati i seguenti parametri:

CmdExec	1024A	IN	Comando di lancio azione per il S.O.
RtnCode	2A	OUT	Return Code (Attualmente non usato) Impostare a Blank

Per questa exit non vengono forniti i sorgenti per l'architettura ACG.

Il parametro **ArcPreEst** è necessario solo nella modalità NetAcross batch. Indica il programma da eseguire per ottenere le informazioni relative all'esecuzione dell'azione che verrà eseguita in base alla selezione di una funzione estesa. Vengono passati i seguenti parametri:

UsrName	10A	IN	Nome dell'utente (UsrPrf)
JobName	10A	IN	Nome del job (terminale virtuale) della

			sessione utente.
OrigVTName	10A	IN	Nome del job (terminale virtuale) della sessione utente che ha richiesto la funzione estesa (# = scelta fatta da menù).
SisInf	10A	IN	Sistema informativo attivo.
MnuIni	10A	IN	Menù iniziale attivo.
LibArch	10A	IN	Libreria di architettura
PrmKpjba	502A	OUT	Kpjba dell'azione
CurLib	10A	OUT	Parametro CurLib per SbmJob.
OutQ	21A	OUT	Parametro OutQ per SbmJob (libr/ogg).
JobQ	21A	OUT	Parametro JobQ per SbmJob (libr/ogg).
JobD	21A	OUT	Parametro JobD per SbmJob (libr/ogg).
LibrList	2750A	OUT	Parametro InLibL per SbmJob.
RtnCode	2A	OUT	Return Code (Attualmente non usato) Impostare a Blank

Per questa exit vengono forniti i sorgenti per l'architettura ACG: NABch2BAP (CLLE), NABch2BA (RPGLE).

UTILITY per Programmatori

NetAcross utilizza i file video 5250 per generare delle pagine Html per poi gestirle a tempo di esecuzione all'interno di un browser.
Oltre agli Html vengono mantenute delle strutture aggiuntive (i files NADDS* all'interno della libreria NETAF) dove vengono memorizzate le varie informazioni che contiene un file video (nomi dei campi, indicatori, keywords, ecc.). A sua volta all'interno dei files NADDS* ce n'è una parte dedicata ai singoli formati video (NADDS01F e NADDS02F), una parte dedicata alla composizione dei formati (NADDS04F e NADDS06F) e una parte dedicata alle traduzioni in lingua (NADDS07F).
Le utility che seguono assistono il programmatore ogni qualvolta si ha bisogno di copiare, cancellare, ridenominare file video perché queste provvedono a tenere allineati tutti gli oggetti o archivi collegati. In particolare queste utility possono essere utilizzate per generare gli oggetti necessari nella creazione delle release dei pacchetti software.

UTILITY per Copia Singolo File Video, NADDS* e Html (cmd NACPYDSP)

Questa utility permette di copiare un file video. Vengono duplicati anche tutti gli oggetti NetAcross correlati. In particolare questa utility è utile quando un programmatore deve modificare un file video ma si vuole fare la modifica ed i test in un ambiente fuori linea (libreria di test). Al termine si eseguirà il comando con i parametri rovesciati per portare in linea il file video con le modifiche eseguite. Attenzione: gli eventuali dati presenti della destinazione vengono sovrascritti.
Il nome dell'utility è **NACPYDSP** e va richiamato come comando.
Vengono richiesti i seguenti parametri:
FILE è il nome del file video da copiare. Campo alfabetico da 20 caratteri, di cui i primi 10 sono l'oggetto ed i secondi 10 sono la libreria.
NEWFILE è il nome con cui duplicare il formato video. Campo alfabetico da 20 caratteri, di cui i primi 10 sono l'oggetto ed i secondi 10 sono la libreria.
CPYDSP specifica se duplicare il file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
CPYDDS specifica se duplicare i file NADDS01F/02F (DDS del file video). Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
CPYCOLL specifica se duplicare gli archivi NADDS04F/06F (collegamento campi video/HTML). Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
CPYHTML specifica se duplicare le pagine Html collegate al file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
FROMROOT è il nome dell'istanza NetAcross da cui copiare i dati del file video. Campo alfabetico da 10 caratteri.
FROMAPPL è il nome dell'applicazione da cui copiare i dati del file video. Campo alfabetico da 10 caratteri.
FROMPERS è il nome della personalizzazione da cui copiare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri.
TOROOT è il nome dell'istanza NetAcross in cui copiare i dati del file video. Campo alfabetico da 10 caratteri. Se inserito *ORIGROOT vale l'istanza origine.
TOAPPL è il nome dell'applicazione in cui copiare i dati del file video. Campo alfabetico da 10

caratteri. Se inserito *ORIGAPPL vale l'applicazione origine.
TOPERS è il nome della personalizzazione in cui copiare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri. Se inserito *ORIGPERS vale la personalizzazione origine. Se inserito *NONE vale come senza personalizzazione (tutti spazi).

UTILITY per Copia File Video, NADDS* e Html di un'intera libreria (cmd NACPYLIB)

Questa utility permette di copiare tutti i file video di una libreria in un'altra libreria. Vengono duplicati anche tutti gli oggetti NetAcross correlati. In particolare questa utility è utile per integrare le procedure di salvataggio delle librerie di un'applicativo di una software house con gli oggetti NetAcross che saranno necessari per il runtime dell'utente finale. Al momento del ripristino dell'applicativo sul computer dell'utente questa utility potrà essere integrata alle procedure di installazione per il ripristino degli oggetti NetAcross. Attenzione: la libreria destinazione se esistente viene sovrascritta.
Il nome dell'utility è **NACPYLIB** e va richiamato come comando.
Vengono richiesti i seguenti parametri:
FROMLIB è il nome della libreria da copiare. Campo alfabetico da 10 caratteri.
TOLIB è il nome della libreria in cui copiare. Campo alfabetico da 10 caratteri.
CPYLIB specifica se duplicare tutti gli oggetti della libreria (*YES) oppure no (*NO). La copia avviene tramite Save e Restore, per cui vengono preservati gli attributi degli oggetti. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
CPYDDS specifica se duplicare i file NADDS01F/02F (DDS del file video). Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
CPYCOLL specifica se duplicare gli archivi NADDS* collegati ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
CPYHTML specifica se duplicare le pagine Html collegate ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
FROMROOT è il nome dell'istanza NetAcross da cui copiare i dati del file video. Campo alfabetico da 10 caratteri.
FROMAPPL è il nome dell'applicazione da cui copiare i dati del file video. Campo alfabetico da 10 caratteri.
FROMPERS è il nome della personalizzazione da cui copiare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri.
TOROOT è il nome dell'istanza NetAcross in cui copiare i dati del file video. Campo alfabetico da 10 caratteri. Se inserito *ORIGROOT vale l'istanza origine.
TOAPPL è il nome dell'applicazione in cui copiare i dati del file video. Campo alfabetico da 10 caratteri. Se inserito *ORIGAPPL vale l'applicazione origine.
TOPERS è il nome della personalizzazione in cui copiare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri. Se inserito *ORIGPERS vale la personalizzazione origine. Se inserito *NONE vale come senza personalizzazione (tutti spazi).
CRTLIB specifica se creare la directory corrispondente alla libreria destinazione (nella struttura dell'istanza NetAcross). Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.

UTILITY per Cancellazione Singolo File Video, NADDS* e Html (cmd NADELDSP)

Questa utility permette di cancellare un file video e tutti gli oggetti NetAcross correlati. In particolare questa utility è utile per cancellare un file video dopo l'utilizzo dell'utility di duplicazione.
Il nome dell'utility è **NADELDSP** e va richiamato come comando.
Vengono richiesti i seguenti parametri:
FILE è il nome del file video da cancellare. Campo alfabetico da 20 caratteri, di cui i primi 10 sono l'oggetto ed i secondi 10 sono la libreria.
DELDSP specifica se cancellare il file video (oggetto di tipo *FILE). Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
DELDDS specifica se cancellare gli archivi NADDS* (specifici dei formati) collegati al file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
DELCOLL specifica se cancellare gli archivi NADDS* (dei collegamenti dei formati) collegati al file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
DELHTML specifica se cancellare le pagine Html collegate al file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
ROOT è il nome dell'istanza NetAcross in cui cancellare i dati del file video. Campo alfabetico da 10 caratteri.
APPL è il nome dell'applicazione in cui cancellare i dati del file video. Campo alfabetico da 10 caratteri.
PERS è il nome della personalizzazione in cui cancellare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri.

UTILITY per Cancellazione File Video, NADDS* e Html di una libreria (cmd NADELLIB)

Questa utility permette ad un programmatore di cancellare tutti i file video da una libreria e tutti gli oggetti NetAcross correlati.
Il nome dell'utility è **NADELLIB** e va richiamato come comando.
Vengono richiesti i seguenti parametri:
LIBR è il nome della libreria da cancellare. Campo alfabetico da 10 caratteri.
DELLIB specifica se cancellare tutti gli oggetti della libreria (oltre quindi ai formati video). Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
DELDDS specifica se cancellare gli archivi NADDS* (specifici dei formati) collegati ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
DELCOLL specifica se cancellare gli archivi NADDS* (dei collegamenti dei formati) collegati ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
DELHTML specifica se duplicare le pagine Html collegate ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
ROOT è il nome dell'istanza NetAcross in cui cancellare i dati del file video. Campo alfabetico da 10 caratteri.
APPL è il nome dell'applicazione in cui cancellare i dati del file video. Campo alfabetico da 10 caratteri.

PERS è il nome della personalizzazione in cui cancellare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri.

UTILITY per Ridenom. Singolo File Video, NADDS* e Html (cmd NARNMDSP)

Questa utility permette di ridenominare un file video e tutti gli oggetti NetAcross correlati.
Il nome dell'utility è **NARNMDSP** e va richiamato come comando.
Vengono richiesti i seguenti parametri:
FILE è il nome del file video da ridenominare. Campo alfabetico da 20 caratteri, di cui i primi 10 sono l'oggetto ed i secondi 10 sono la libreria.
NEWFILE è il nome oggetto con cui ridenominare il formato video. Campo alfabetico da 10 caratteri.
RNMDSP specifica se ridenominare il file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
RNMDDS specifica se ridenominare il contenuto degli archivi NADDS* (specifici dei formati) collegati ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
RNMCOLL specifica se ridenominare il contenuto degli archivi NADDS* (dei collegamenti dei formati) collegati al file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
RNMHTML specifica se ridenominare le pagine Html collegate al file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
ROOT è il nome dell'istanza NetAcross in cui ridenominare i dati del file video. Campo alfabetico da 10 caratteri.
APPL è il nome dell'applicazione in cui ridenominare i dati del file video. Campo alfabetico da 10 caratteri.
PERS è il nome della personalizzazione in cui ridenominare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri.

UTILITY per ridenom. una libreria con File Video, NADDS* e rel. Html (cmd NARNMLIB)

Questa utility permette di ridenominare un intera libreria e sincronizzare tutti gli oggetti NetAcross correlati a tutti i file video contenuti.
Il nome dell'utility è **NARNMLIB** e va richiamato come comando.
Vengono richiesti i seguenti parametri:
FROMLIB è il nome della libreria originale. Campo alfabetico da 10 caratteri.
TOLIB è il nome della libreria risultante. Campo alfabetico da 10 caratteri.
RNMLIB specifica se ridenominare l'oggetto libreria. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
RNMDDS specifica se ridenominare il contenuto degli archivi NADDS* (specifici dei formati) collegati ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
RNMCOLL specifica se ridenominare gli archivi NADDS* (dei collegamenti dei formati) collegati ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
RNMHTML specifica se ridenominare le pagine Html collegate ai file video. Campo alfabetico da 4 caratteri. Valori validi *YES o *NO.
ROOT è il nome dell'istanza NetAcross in cui ridenominare i dati del file video. Campo alfabetico da 10 caratteri.

APPL è il nome dell'applicazione in cui ridenominare i dati del file video. Campo alfabetico da 10 caratteri.
PERS è il nome della personalizzazione in cui ridenominare i dati del file video. Se viene inserito spazi non viene ricercata una personalizzazione. Campo alfabetico da 10 caratteri.

Configurazioni

Il file config.xml

Il file config.xml è la configurazione di un’istanza net-Across, contiene informazioni sia di conversione dei display che di runtime.
Si trova nella sottocartella /config di una document root, per default la /netacross/config/config.xml è la sua posizione principale, e deve esistere almeno una versione di questo file per avviare il sottosistema NETACROSS e poter utilizzare i servizi.

Livello	Voce	Obb/ Fac	Significato	Valori
1	<netacross><default>	O	Inizio struttura	
3	<Pages_Root>	F	Contiene la cartella padre per le conversioni degli HTML	(default: NETACROSS)
3	<TCP_Config>	O	Area configurazione TCP	
4	<Server_IP>	O	Deve contenere l’IP dell’ars su cui è attivo il servizi	(es.: 127.0.0.1)
4	<PortRange>	O	Area in cui si identificano le porte TCP/IP disponibili ad essere usate per le sessioni batch di NetAcross. Devono essere usate le porte alte del TCP/IP non usate da altre applicazioni, altrimenti si genereranno errori durante il tentativo di utilizzo.	
5	<From>	O	Limite di porta inferiore	(es.: 30001)
5	<To>	O	Limite di porta superiore	(es.: 30999)
3	<NbrSrvJvaFix>	F	Numero di server Java attivi in avvio automatico nel sottosistema NETACROSS	(default: 1)
3	<NbrSrvJvaVar>	F	Numero (max) di server Java attivati in caso di richieste contemporanee	(default: 1)
3	<RunTime>	O	Area configurazioni runtime	
4	<GzipDefault>	F	Abilitazione generale della compressione dati di pagina in trasmissione/ricezione da/per browser.	(default: N) valori: Y,N
4	<MaxTermNum>	F	Stabilisce quante sessioni browser possono essere aperte da un utente se è specificato un nome terminale nella sua configurazione utente. Viene aggiunto un suffisso per ogni sessione aperta.	(default: 3) valori: 1 - 9

Livello	Voce	Obb/ Fac	Significato	Valori
4	<LanguageUse>	F	Abilitazione delle conversioni in lingua per le costanti delle videate	(default: N) valori: Y,N
4	<UsrCertName>	F	Utente validato da certificato per riconoscimento client, se non presente non abilita la funzione. Deve corrispondere ai primi 8 caratteri della variabile Common Name del certificato client.	
4	<JScripPers>	F	Indica il percorso relativo a Root di un file *.js contenente funzioni javascript personalizzate da aggiungere a tutte le pagine dinamiche	(es.: /pers/ funzioni.js)
4	<showTerminalMsg>	F	Visualizza i messaggi per le sessioni terminale interattivo. Si riferisce alla visualizzazione della coda *EXT se ci sono messaggi. Vale solo per modalità interattiva.	default: Y valori: Y,N
4	<showWindows>	F	In presenza di un formato window DDS crea una window in Html oppure a pagina piena.	default: N valori: Y,N
4	<showHTML>	F	Permette di vedere il sorgente HTML della pagina.	default: N valori: Y,N
4	<GestWriteComp>	F	Tiene traccia delle emissioni di Windows con sola Write e le presenta alla prima Exfmt.	default: N valori: Y,N
4	<Tr_Title_Sep>	F	Definisce se inserire dopo la riga del titolo nella pagina Html una riga bianca di spaziatura da 10 pixel	default: Y valori: Y,N
4	<Titolo01_Center>	F	Definisce se il titolo della pagina deve avere i campi centrati orizzontalmente.	default: Y valori: Y,N
4	<ClassProtect>	F	Permette di definire una classe o uno stile diverso da attribuire ai campi di input quando diventano protetti. In particolare è utile quando si vuole dare ad un campo protetto un aspetto simile ad un campo non protetto. Se la keyword non viene inserita i campi protetti saranno visualizzato come i campi solo output.	(es.: input01 oppure “color: #868788;”
4	<ClassError>	F	Permette di definire una classe o uno stile diverso da attribuire ai campi quando sono in errore.	default: error01
4	<KeyFView>	-	!DEPRECATA! Sostituita dal corrispondente valore sul commands_defaults.js	
4	<MenuCallCmd>	F	Abilita la chiamata diretta ad azione/comando dal menù ad albero di architettura	default: N valori: Y,N

Livello	Voce	Obb/ Fac	Significato	Valori
4	<AutoHyperLink>	F	Specifica se trasformare i campi di output che fanno riferimento ad un URL in hyperlink. L'URL deve iniziare per http://, ftp://, file://.	default: N valori: Y,N
4	<SfpSepDft>	F	Separatore delle coppie scelta/ descrizione per la keyword *SFP	default: = valori: un carattere
4	<JobDefArea>	F	Area per l'instradamento dei lavori sessione in modalità batch NetAcross.	
5	<JDA_Rule>	F	Area dei ruoli associati a device. Sono possibili un massimo di 20 JDA_Rule.	
6	<JDAR_Device>	O	Nome dei devices a cui si riferisce quest'area. Si può inserire un valore specifico oppure un nome parziale con nome* oppure completo con *.	
6	<JDAR_JobQ>	F	Nome della coda lavori in cui inserire la sessione. Nella forma libreria/coda o coda.	default: QINTER
6	<JDAR_RtgDta>	F	Routing Data per instradare il lavoro. Max 5 caratteri.	default: QCMDI
4	<UsrMsgPolling>	F	Area della gestione messaggi operatore	
5	<UMP_Active>	F	Abilitazione del pulsante gestione messaggi. Permette di accedere alla coda messaggi utente e della sessione terminale. I parametri di funzionamento sono definiti nel file commands_defaults.js	default: N valori: Y,N
4	<PasswordList>	F	Area del controllo password per utenti controllati da Validation List. E' necessario che sia configurata la funzione estesa con oggetto PASSWORD ed estensione PWD.	
5	<Password>	F	Area che identifica la validation list con la scadenza password attivata	
6	<PasswordName>	F	Nome della validation list nella forma NETAF/nome_validation	
6	<GgPwdValid>	F	Giorni di validità della password espresso con 3 cifre	(es.: 060)
6	<GgPwdAlert>	F	Giorni precedenti alla scadenza a cui l'utente è avvisato della prossima scadenza, espresso con 3 cifre	(es.: 015)
6	<GgPwdMinCh>	F	Lunghezza minima della password in caratteri, espresso con 3 cifre. Valore valido solo per validation list	(es.: 005)

Livello	Voce	Obb/ Fac	Significato	Valori
4	<InterPwdChk>	F	Per utilizzo INTerattivo, permette di ignorare la password di un USRPRF usando il valore speciale *CURRENTUSER	Y/N default è 'Y' la password deve essere espressain VLDL
4	<Kerberos>	F	Area autenticazione per accessi tramite Single Signon (Kerberos)	
5	<LdapURL>	F	Identifica il server in protocollo LDAP che fornisce l'elenco degli utenti EIM	es.: ldap://172.31.8.247/EIM
5	<LdapUsr>	F	Identifica l'utente LDAP con cui richiedere l'elenco utenti	cn=administrator
5	<LdapPwd>	F	Identifica la password dell'utente <LdapUsr>	
5	<TgtDomain>	F	Identifica il dominio LDAP a cui richiedere l'elenco utenti	es.: AS400GBI.CVS.LAN
4	<ClassesWithAttribute>	F	Area delle classi CSS sensibili agli attributi del video. Le classi dovranno essere definite nel file css associato alla sessione, con i suffissi previsti per attributo, vedi la tabella	
5	<Class>	F	Valore classe che verrà elaborato in base agli attributi attivi sul campo	(es.: const)
4	<Autoconvert>	F	Abilitazione alla conversione a runtime	default: N valori: Y=abili con conferma N=disabilitato F=senza conferma T=tutti senza conf.
4	<AutoConvUsers>	F	Area degli utenti abilitati alla conversione, sono validi sia utenti di sistema che utenti di validation list	
5	<AC_User>	F	Valore del nome utente	
4	<WidthMultInfo>	F	Area elenco di classi per l'assegnazione delle ampiezze orizzontali fisse dei caratteri dei campi di input e di output. Se non specificato il browser assegna l'ampiezza orizzontale in modo automatico: sulla base della size per i campi di Input (+1 carattere extra ogni 8), sulla base dei caratteri presenti per i campi di output. Con questa opzione attivata i campi non fluttuano sulla base del contenuto, però in orizzontale potranno occupare più spazio.	

Livello	Voce	Obb/ Fac	Significato	Valori
5	<WidthMultCFac>	F	Valore della classe=valore dell'ampiezza orizzontale in pixel di un carattere. Valutare sulla base del carattere usato nella classe. Non occorre specificare le classi derivate, cioè quelle determinate dall'aggiunta dell'attributo campo. Ad esempio basterà inserire la classe CONST e non le sue derivate CONSTRI o CONSTHI create per rappresentare il reverse image o l'alta intensità.	
4	<DynamicCss>	F	Area elenco per l'assegnazione di CSS alle pagine Html in modo dinamico in funzione dell'applicazione, della libreria dei file video, dell'utente o della dimensione della finestra del browser. Vedere spiegazione dettagliata nel capitolo Gestione CSS Variabili in questo manuale.	
5	<CssAppl>	F	Abilita o disabilita la funzione di CSS dinamico per la modalità applicazione, libreria	default: N valori: Y,N
5	<CssList>	F	Abilita o disabilita la funzione di CSS dinamico per la modalità allargamento o restringimento in funzione della dimensione della finestra del browser.	default: N valori: Y,N
5	<CssListDet>	F	Area elenco dei CSS per la modalità allargamento o restringimento in funzione della dimensione della finestra	
6	<CssScreen>	F	Singoli CSS per la modalità allargamento o restringimento in funzione della dimensione della finestra del browser.	es.: Width in px; Height in px; 80 o 132; <suffisso css>; Coefficiente add/sub
4	<Tr_Dyn>	F	Le righe vuote vengono rappresentate nell'ordine di una per ogni gruppo contiguo di righe. Questo per non uscire dai video con poca definizione. Questa chiave permette di abilitare la rappresentazione delle righe omesse per questo motivo	default: N valori: Y,N
5	<Tr_Dyn_Height>	F	Questa chiave esprime il numero di pixel con cui rappresentare le righe vuote omesse. E' abilitata se è immesso Y nella chiave precedente.	(es.: 20)
4	<ObjMime>	F	Per gli oggetti distribuiti tramite Alias e se nell'Alias non viene indicata un tipo applicazione (directory con dati misti) con questa chiave si possono definire le associazioni da fornire al browser.	

Livello	Voce	Obb/ Fac	Significato	Valori
5	<ObjMimeDet>	F	Coppia estensione/tipo Mime per ogni tipo file contenuto nella directory puntata dall'Alias. Inserire l'estensione comprensiva del punto, uguale e il tipo Mime così come definito standard. A livello di browser dal tipo si ricaverà l'estensione e da questa l'applicazione da attivare per la visualizzazione.	(es.: pdf=application/pdf doc=application/ msword xls=application/ vnd.ms-excel
4	<MsgError_2L>	F	Indica se in caso di messaggio di errore deve essere visualizzato anche il testo di secondo livello (se presente).	default: N valori: Y,N
4	<HelpRoot>	F	Indica la cartella che contiene il testo di Help indicato con la keyword *HLP. La cartella deve essere pubblicata dal server Http.	
4	<ArcList>	F	Area elenco architetture con menù ad albero	sono previste fino a 9 elementi di dettaglio
5	<ArcDef>	F	Area architettura per menù ad albero, elemento di dettaglio	
6	<ArcName>	F	Nome identificativo dell'architettura	testo libero
6	<ArcPgm>	F	Nome programma iniziale come scritto nel programma iniziale del profilo utente, è la chiave per riconoscere l'architettura	(es.: LIBRERIA/ PROGRAMMA)
6	<ArcUsrNfo>	F	Exit per info su utente collegato	(es.: LIBRERIA/ PROGRAMMA)
6	<ArcMnuLst>	F	Exit per recuperò lista dei menù e azioni	(es.: LIBRERIA/ PROGRAMMA)
6	<ArcSisLst>	F	Exit per recupero lista dei sistemi informativi abilitati per l'utente	(es.: LIBRERIA/ PROGRAMMA)
6	<ArcVidMnu>	F	Nome del file video gestione menù architettura	(es.: LIBRERIA/ FILEVIDEO.FORMATO)
6	<ArcHomeKey>	F	Tasto per tornare al menù iniziale del video che gestisce l'architettura, solo per modalità INT (interattiva)	Fxx
6	<ArcEndKey>	F	Tasto funzione per la SIGNOFF, solo per la modalità interattiva	default: F23
6	<ArcFldCho>	F	Campo del video di architettura per inserimento opzione	nome_campo
6	<ArcPreSbm>	F	Exit per lancio delle zioni selezionate dal menù ad albero	LIB/PGM

Livello	Voce	Obb/ Fac	Significato	Valori
6	<ArcPreCal>	F	Exit per esecuzione di un programma prima del lancio di ogni azione	LIB/PGM
6	<ArcPosCal>	F	Exit per esecuzione di un progamma dopo la fine di una azione	LIB/PGM
6	<ArcPreEst>	F	Exit per lancio dei comandi associati alle funzioni estese	LIB/PGM
3	<HtmlGenerator>	O	Area delle informazioni di conversione HTML	
4	<MiniRollOn>	F	Specifica come rappresentare i pulsanti di roll sul subfile non esteso. Y=direttamente sul subfile N=nella pulsantiera dei comandi. Nel caso di due subfile nella stessa pagina vale sempre Y.	default: N valori: Y,N
4	<Template>	O	Percorso relativo alla ROOT di un file html che fa da scheletro per tutti gli html convertiti. Nel caso di funzione *COM il file cercato è template.COM.html. Se sono presenti sottocartelle con nome LIBRERIA con sottocartella i NOMEFILEVIDEO il file template viene ricercato prima qui.	(es.: /config/ template.html)
4	<Primitive>	O	Percorso relativo alla ROOT che contiene le tag di riferimento per la creazione dell'html	(es.: /config/ primitive.xml)
4	<KeyFSplit>	F	Determina se ricercare il testo dei tasti di comando nelle pagine 5250 come campi separati oppure se è presente un unico campo con all'interno le varie costanti. Collegata alla keyword KeyFunctions	default: N valori: Y,N
4	<SflHdrDft>	F	Numero di righe per salire nella ricerca della testata da tagliare per i subfile con drop. Modificabile a livello di file video con la keyword *HDR	default: 2 valori: 1 - 9
4	<RadioDefault>	F	Determina quale tipo di radio button generare se l'applicazione ha valutato che è opportuno trasformare un campo in radio button. Il tipo di menù da generare può anche essere condizionato con la keyword NA*MNU.	default: O valori: O= orizzontale V=verticale
4	<KeyFunctions>	O	Area di definizione delle espressioni regolari usate per recuperare i tasti funzionali dal video	

Livello	Voce	Obb/ Fac	Significato	Valori
5	<Pattern> (alternativo a <KP>)	F	Espressione regolare che riconosce il tasto funzione in una costante del video. Possono essere inserite più righe. Le regole di composizione sono quelle delle espressioni regolari dell'AS/400, compatibili con lo standard C UNIX In alternativa per esprimere il delimitatore comando/testo si può esprimere con la sintassi KP che segue.	(es.: F[0-9]+[0-9]*.+)
5	<KP>	F	Area che raggruppa le espressioni regolari per recuperare le descrizioni dei tasti funzioni del video (con delimitatore). **Questa sintassi si può esprimere in alternativa alla sintassi Pattern precedente**. In più questa permette di esprimere il delimitatore comando/testo. Possono essere inserite più righe	
6	<Pattern>	F	Espressione regolare che riconosce il tasto funzione in una costante del video.Le regole di composizione sono quelle delle espressioni regolari dell'AS/400, compatibili con lo standard C UNIX	(es.: F[0-9]+[0-9]*.+)
6	<Delim>	F	Delimitatore fra il tasto di comando ed il testo che rappresenta.	(es.: =)
4	<Subfile>	F	Area che raggruppa le opzioni di generazione relative ai subfile	
5	<SplitMinChars>	F	Definisce l'ampiezza minima in caratteri perché una costante della testata del subfile venga divisa in funzione delle colonne del subfile stesso	default: 40
5	<ToXLS>	F	Definisce il comportamento di default per l'abilitazione dei subfile al collegamento con fogli di calcolo. Vale solo per la modalità batch.	default: N valori: Y,N
5	<XlsAlias>	F	Nome dell'alias dove scrivere i file temporanei di fogli di calcolo collegati al pulsante xls sui subfile e visualizzabili lato client. Vale solo per la modalità batch.	(es.: xls)
5	<XlsClass>	F	Nome della classe CSS che verrà usata per costruire il pulsante xls visualizzabile sui subfile. Vale solo per la modalità batch.	
4	<ClassNumeric>	F	Permette di assegnare una classe particolare ai campi numerici. Utilizzando una classe con passo fisso permette di allineare correttamente a destra le cifre contenute.	(es.: const)

Livello	Voce	Obb/ Fac	Significato	Valori
3	<PDF_Config>	F	Area che raggruppa le opzioni di generazione dei PDF dalla gestione Spool (NASPLFP) fornita nelle Funzioni Estese – Utility. Usare solo se c'è un gestore PDF che lavora sulle code di stampa. Se non si usa viene usato il supporto NetAcross SP2PDF.	
4	<PDFOutQ>	F	Nome della coda di output alla quale è collegata la generazione dei files PDF.	LIBRERIA/NOMECODA
3	<MainDocAlias>	F	Nome dell'alias per il reperimento dei documenti dalla funzione Gestione Documenti (NAWRKLNK) fornita nelle Funzioni Estese - Utility.	(es.: nadocs)

Le classi 'hard coded'

Durante la fase di conversione della videata in HTML , il generatore di html applica delle classe a livello di riga e a livello di campo.

Nello specifico per le righe:

La prima riga dello schermo avrà classe **"titolo01"**
La prima riga di ogni formato avrà classe **"titolo02"**
Le righe di un form avranno classe **"titolo04"**
Le righe dispari di un subfile (no *SFX) avranno classe **"const01"**
Le righe pari di un subfile (no *SFX) avranno classe **"const02"**
Lo sfondo dello schermo avrà classe **"sfondo01"**
Il piede di ogni videata avrà classe **"fondo01"**

Per i campi:

Un campo in errore (indicatore accesso) avrà classe **"error01"**
Secondo le specifiche di default del config.xml:
Un campo alfabetico ha classe **"normal"**
Un campo numerico ha classe **"const"**

Le classi con attributi variabili

I campi video a livello di DDS possono prevedere degli attributi di visualizzazione (alta intensità, sottolineato, ecc.) e di colore (rosso, blu, ecc.). Se il campo visualizzato ha una classe elencata nella voce **<ClassesWithAttribute>** l'attributo viene rimappato in una classe del CSS aggiungendo un suffisso predefinito al nome della classe permettendo di simulare a livello di

Html l'effetto che il campo aveva a livello di 5250.
Sarà compito dell'utente prevedere nel CSS le classi per ogni attributo utilizzato.

In questo modo è possibile alterare la grafica di un campo gestendo con gli indicatori a programma la visualizzazione di determinati attributi. La risultante grafica definita nel CSS è completamente indipendente dai colori definiti sul display.

Codici Attributi 5250

Colore/ Opzione	NO	RI	UL	RU	BL	RB	UB
G reen	x'20'	x'21'	x'24'	x'25			
W hite	x'22'	x'23'	x'26				
R ed	x'28'	x'29'	x'2C'	x'2D'	x'2A'	x'2B'	x'2E'
T orquise	x'30'	x'31'	x'34'	x'35'			
Y ellow	x'32'	x'33'	x'36'				
P ink	x'38'	x'39'	x'3C'	x'3D'			
B lue	x'3A'	x'3B'	x'3E'				

Suffissi CSS da aggiungere alla classe

Colore/ Opzione	NO	RI	UL	RU	BL	RB	UB
G reen	G	GRI	GUL	GRU			
W hite	W	WRI	WUL				
R ed	R	RRI	RUL	RRU	RBL	RRB	RUB
T orquise	T	TRI	TUL	TRU			
Y ellow	Y	YRI	YUL				
P ink	P	PRI	PUL	PRU			
B lue	B	BRI	BUL				

Il file commands_defaults.js

Il file commands_defaults.js è agganciato all'html start.html che fa da pannello di struttura per i tasti funzionali, per i dati personalizzati in testata e per le funzioni di sessione.

```
var DefaultX = 100;                                  // variabile di riferimento
var DefaultY = 30;                                   // variabile di riferimento
var COMMANDS = 30;                                   // numero massimo di comandi rappresentabili
var PANEL_X_SIZE = 100;                              // n.pixel per ampiezza orizzontale pannello
var PANEL_Y_SIZE = 919;                              // n.pixel per ampiezza verticale pannello
var PANEL_BG_IMAGE = "/images/sam_background.gif";   // immagine di sfondo del pannello
```

```
var PANEL_BOTTOM_X_SIZE = DefaultX;                    // n.pixel per amp.orizz. immagine del pannello
var PANEL_BOTTOM_Y_SIZE = DefaultY;                    // n.pixel per amp.vert. immagine del pannello
var PANEL_BOTTOM_IMAGE = "/images/box_bottom_.jpg"; // immagine di chiusura pannello
var PANEL_SCROLLER = 'N';                              // freccie di scorrimento del pannello
var CMD_X_SIZE = DefaultX;                             // n.pixel per ampiezza orizzontale comando
var CMD_Y_SIZE = DefaultY;                             // n.pixel per ampiezza verticale comando
var CMD_IMAGE = "/images/box_head_.gif";               // immagine per il comando abilitato
//var CMD_DISABLED_IMAGE = "/images/pixel.gif";        // immagine vuota per il tasto disabilitato
var CMD_DISABLED_IMAGE = "/images/box_bg_.gif";        // immagine per il tasto disabilitato
var CMD_Y_UTIL_SIZE = DefaultY - 6;                    // n.pixel spazio utile verticale del testo
var CMD_X_UTIL_SIZE = DefaultX - 20;                   // n.pixel spazio utile orizzontale del testo
var CMD_FONT_FAMILY = "Helvetica";                     // font family del testo
var CMD_FONT_SIZE = 11;                                // font size in points del testo
var CMD_FONT_COLOR = "#333333";                        // colore del testo
var TOP_X_SIZE = DefaultX;                             // numero di pixel orizzontali per immagine top
var TOP_Y_SIZE = 19;                                   // numero di pixel verticali per immagine top
var TOP_IMAGE = "/images/box_top.gif";                 // immagine per il top
var BOTTOM_X_SIZE = DefaultX;                          // n.pixel orizzontali per immagine bottom
var BOTTOM_Y_SIZE = 30;                                // n.pixel verticali per immagine bottom
//var BOTTOM_IMAGE = "/images/box_bottom_.jpg";        // immagine chiusa per il bottom
var BOTTOM_IMAGE ="/images/pixel.gif";                 // immagine trasparente per il bottom
var ROLLUP_IMAGE = "/images/rollup_.gif";              // immagine per Pagina Avanti
var ROLLDOWN_IMAGE = "/images/rolldown_.gif";          // immagine per Pagina Indietro
var CMD_VIEW = "V";                                    // comandi in verticale "V" o orizzontale "O"
Var DELETE_BLANK_KEYF = "N"                            // cancella tasti comando senza descrizione

var TREE_SISINF_VISIBLE = "Y";                         // visibilità scelta sistemi informativi
var TREE_BGCOLOR = '#FFFFFF';                          // colore di sfondo
var TREE_FOLDEROPEN = '/scripts/xdom/RC/folder_open.gif'    // icona menù aperto
var TREE_FOLDERCLOSE = '/scripts/xdom/RC/folder_close.gif'; // icona menù chiuso
var TREE_ITEM = '/scripts/xdom/RC/eye.gif';                 // icona azione
var TREE_ITEM_HSIZE = 20;                              // amp.verticale in pixel menù/azioni
var TREE_ITEM_FONTFAMILY = "Helvetica";                // tipo font azioni
var TREE_ITEM_FONTSIZE = "13px";                       // ampiezza font azioni
var TREE_ITEM_FONTWEIGHT = "bolder";                   // peso font azioni
var TREE_ITEM_FONTCOLOR = "#012345";                   // colore font azioni
var TREE_ITEM_SELECT_FONTFAMILY ="Arial";              // Attributo riga quando puntata
var TREE_ITEM_SELECT_BGCOLOR ="#ABCDEF";               // Attributo riga quando puntata
var TREE_ITEM_SELECT_FONTCOLOR ="black";               // Attributo riga quando puntata
var TREE_FOLDER_FONTFAMILY = "Helvetica";              // tipo font opzioni di menù
var TREE_FOLDER_FONTSIZE = "14px";                     // ampiezza font opzioni di menù
var TREE_FOLDER_FONTWEIGHT = "bolder";                 // peso font opzioni di menù
var TREE_FOLDER_FONTCOLOR = "#654321";                 // colore font opzioni di menù
var TREE_COLLAPSE_BUTTON = "Y";                        // tasto reset menù Si/No
var TREE_COLLAPSE_BUTTON_TEXT = "Reset Menu'";         // tasto reset descrizione
var TREE_COLLAPSE_BUTTON_RIGHT = "100px";              // tasto reset posizione
//var TREE_COLLAPSE_BUTTON_LEFT = "100px";             // tasto reset posizione
//var TREE_COLLAPSE_BUTTON_TOP = "1px";                // tasto reset posizione
var TREE_COLLAPSE_FONTSIZE = "13px";                   // tasto reset ampiezza font
var TREE_COLLAPSE_FONTCOLOR = "black";                 // tasto reset colore
var TREE_COLLAPSE_FONTFAMILY = "Arial";                // tasto reset font
var FRAME_TITLE_ONCHOICE = '*DES - net-across © 2006'       // Valori speciali:
                                                            //'*DES' descrizione della azione in
                                                            // corso
                                                            //'*SIS' descrizione sistema informativo

var FRAME_TITLE_DEFAULT = "net-across © Computer Var Services spa"; // default per titolo del frame
var CALLCMD_ACTIVE= "Y"                                     // Abilitazione
var CALLCMD_SIZE = 4;                                       // n.caratteri chiamata ad azione
//var CALLCMD_RIGHT = "300px";                              // Posizione
//var CALLCMD_LEFT = "200px";                               // Posizione
//var CALLCMD_TOP = "0px";                                  // Posizione
//var CALLCMD_FONTSIZE = "13px";
//var CALLCMD_FONTWEIGHT = "bolder"; /* default */
//var CALLCMD_FONTCOLOR = "#000000"; /* default */
//var CALLCMD_UPPERCASE = true; /* default */
```

```
var MSG_BUTTON_WIN = 'TOP'; // [TOP MAIN COMMANDS NO]       // Frame di riferimento
var MSG_BUTTON_POSX = "50%"; /* default */                  // Posizione X in % o px
var MSG_BUTTON_POSY = "5px"; /* default */                  // Posizione Y in % o px
var MSG_BUTTON_WIDTH = "100px"; /* default */
var MSG_BUTTON_HEIGHT = "30px"; /* default */
//var MSG_BUTTON_TITLE = " Messaggi ";  /* default */
//var MSG_BUTTON_BACKGROUND = "#CDAFEA"; /* default */
var DELETE_BLANK_KEYF = "Y"; /* default */                  // non visualizza i tasti senza
                                                            // descrizione, ma i tasti rimangono
                                                            // attivi [valori Y/N]
var KEYFVIEW = "N"; /* default */                           // [valori Y/N] visualizza il tasto
                                                            // funzione insieme alla descrizione nel
                                                            // testo del comando
var SFX_NRIG_COLUMN = "NONE" /* default */                  // [valori RIGHT / LEFT / NONE ]
                                                            // presenza e posizione del conta righe
var BLOCK_FKEY = "Y";                                       // Blocca i tasti funzione non definiti
var SID_ACTIVE = "Y";                                       // attiva i coockie di sessione locale
var RIGHTMENU_MONO_DIM = "SI";                              // menù tasto destro solo verticale
```

La toolbar

All'interno della pagina Html Top.htm che rappresenta la parte alta della pagina e dove di solito si posizionano il marchio aziendale e i pulsanti fissi è possibile attivare i tasti di comando che per l'applicazione hanno un significato costante.
Ad esempio i tasti uscita programma, video precedente, pagina avanti ed indietro, invio, ecc. sono dei candidati ad essere portati in Toolbar.
Questi comandi una volta portati in Toolbar liberano la barra dei comandi standard lasciando solamente i tasti di comando particolari per la pagina.
Per attivare o personalizzare la toolbar si agisce sul codice seguente, inserito nel file Top.htm:

```
<link rel="stylesheet" href="toolbar/natoolbar.css" type="text/css">
<div id="natoolbar">

<button class="natool_button" id="button_Info" onclick="parent.topFrame.showInfo();" title="Info">
        <div class="natool_image" ></div>
        <div class="natool_text">
                Info
        </div>
</button>

<button class="natool_button" id="button_Inquiry" onclick="parent.topFrame._startQuestion();"
title="Search">
        <div class="natool_image" ></div>
        <div class="natool_text">
        Search
        </div>
</button>

<button class="natool_button" id="natoolkey_Help" >
        <div class="natool_image" ></div>
        <div class="natool_text">
        Help
        </div>
</button>

<button class="natool_button" id="natoolkey_Invio" >
        <div class="natool_image" ></div>
        <div class="natool_text">
        Enter
        </div>
</button>
```

```
<button class="natool_button" id="natoolkey_F12" >
        <div class="natool_image" ></div>
        <div class="natool_text">
        Precedente
        </div>
</button>

<button class="natool_button" id="natoolkey_F03" nastatic="true">
        <div class="natool_image" > </div>
        <div class="natool_text">
                Exit
        </div>
</button>

<button class="natool_button" id="natoolkey_RollUp" nastatic="true">
        <div class="natool_image" ></div>
        <div class="natool_text">
        PageDown
        </div>
</button>

<button class="natool_button" id="natoolkey_RollDown" nastatic="true">
        <div class="natool_image" ></div>
        <div class="natool_text">
        PageUp
        </div>
</button>

<button class="natool_button" id="natoolkey_XLS" nastatic="true">
        <div class="natool_image" ></div>
        <div class="natool_text">
        to Excel
        </div>
</button>

<button class="natool_button" id="natoolkey_MSG" nastatic="true">
        <div class="natool_image" ></div>
        <div class="natool_text">
        Messaggi
        </div>
</button>

</div>
```

La toolbar contiene oggetti button che possono avere una descrizione bloccata se **nastatic="true"** oppure possono cambiare la descrizione in base al valore testo del tasto funzionale collegato, l'immagine rimane invece costante.
Tramite il css collegato nella tag link **toolbar/natoolbar.css** è possibile stabilire dimensioni e colori dei pulsanti, le immagini, se il testo sarà visibile o meno, se l'immagine sarà visibile o meno e la posizione della toolbar nel top frame.

Alla toolbar è possibile aggiungere un pulsante di selezione al fine di permettere la attivazione di un foglio di stile (CSS) particolare per la sessione. Ad esempio è possibile attivare diversi fogli di stile se si sono attivate più sessioni contemporaneamente al fine di distinguerle. Resta ferma la possibilità di attribuire automaticamente un foglio di stile all'applicazione, al sistema informativo o all'utente.
I fogli di stile selezionati con questa opzione sono additivi al foglio di stile di base, e cioè modificano solo le classi specificate, lasciando inalterate le altre.

Per attivare questa funzione aggiungere al codice sorgente specificato nelle righe precedenti il codice:

```
<div class="natool_divcss">
  <select id = "natool_selectcss" ></select>
</div>
```

Creare con un normale editor nella directory /netacross/style il file ListUsrCss.txt con il contenuto che segue (esempio):

```
{
cartelle :
["/style/usr/blu.css","/style/usr/Verde.css","/style/usr/Bianco.css"],
descrizioni :
["Sfondo Blu","Sfondo Verde","Sfondo Bianco"]
}
```

Il pulsante di selezione verrà riempito con le costanti elencate in <descrizioni> e quando viene effettuata la selezione in override al foglio di stile di base viene aggiunto il foglio di stile corrispondente alla voce <cartelle>.

I miniroll NEW! novembre 2014

In base al config.xml è possibile visualizzare i pulsanti di paginazione sotto il subfile, tramite due pulsanti chiamati miniroll. Questa opzione non si applica ai subfile con *SFX ma solo a quelli standard.
E' possibile personalizzare la grafica di questi pulsanti utilizzando un file di configurazione: /netacross/config/primitive.xml.
Dentro questo file ci sono le regole generali per la costruzione delle pagine html relative ai formati video, per quanto riguarda le parti dinamiche.
In particolare l'area interessata è:

```
<na_pageroll>
<tr height="16" class="sfondo01">
 <td colspan="">
 <div class="page-roll">
  <img onclick="NetA.setRollDown('<FMT_NAME>')" src="/images/mini_rolldown.gif" class="roller-down" />
  <img onclick="NetA.setRollUp('<FMT_NAME>')" src="/images/mini_rollup.gif" class="roller-up" />
 </div>
 </td>
</tr>
</na_pageroll>
```

l'area <na_pageroll> in questo caso è stata già predisposta con le classi (nella versione rilasciata con il prodotto il contenuto dell'area sarà diverso) .
Naturalmente la parte riguardante il javascript e il nome del formato NON DEVONO essere modificati pena l'errato funzionamento dei miniroll.

Date Aree (DTAARA)

NABUILD in NETAP

Contiene il numero versione di NetAcross installata

NAPTFLVL in NETAP

Contiene la data delle ultime PTF cumulative installate

NADEBUG in NETAF

Se impostata a '1' abilita la joblog dei lavori java del sottosistema NETACROSS e monitoriaza le operazioni dei lavori netacross tramite il file NALOG00F
CHAR 1

NAJOBLOG in NETAF

Se impostata a 'N' evita le joblog dei lavori netacross di sessione e cgi in conseguenza di chiusure , questo velocizza la fare di chiusura delle sessioni ma limita la possibilità di analisi di eventuali errori riscontrati a run-time.
CHAR 1

NAJVAMEM in NETAF

Permette di inserire un valore passato alla JVM per modificare la dimensione della memroia di heap nel caso di elevato consumo di RAM da parte di operazioni java (pf2xlsx)
CHAR 10

NASAVQTEMP in NETAF

Se impostata a '1' permette di salvare il contenuto della QTEMP tra una azione ed un'altra di una sessione ad architettura con menù ad albero, essendo ogni azione un job separato il salvataggio della QTEMP non è automatico. L'opzione può comportare un appesantimento della fase di avvio dell'azione in funzione del numero di oggetti presenti in QTEMP
CHAR 1

NASQLCH1 in NETAF

E' il carattere di separazione (inizio) per identificare le variabili negli SQL delle funzioni *COM
CHAR 1

NASQLCH2 in NETAF

E' il carattere di separazione (fine) per identificare le variabili negli SQL delle funzioni *COM
CHAR 1